

**RECHNERUNTERSTÜTZTE OPTIMIERUNG
DES ABBAUS VON RÜBENSCHWEMMWASSER
MITTELS EINES ANAEROBFILTERS**

INAUGURAL-DISSERTATION

zur

Erlangung der philosophischen Doktorwürde

vorgelegt der

PHILOSOPHISCHEN FAKULTÄT II

der

UNIVERSITÄT ZÜRICH

von

ETTORE DA PRA

von Lozzo di Cadore (Italien)

Begutachtet von

Prof. Dr. R. Bachofen

Zürich 1987

Zentralstelle der Studentenschaft

**RECHNERUNTERSTÜTZTE OPTIMIERUNG
DES ABBAUS VON RÜBENSCHWEMMWASSER
MITTELS EINES ANAEROBFILTERS**

INAUGURAL-DISSERTATION

zur

Erlangung der philosophischen Doktorwürde

vorgelegt der

PHILOSOPHISCHEN FAKULTÄT II

der

UNIVERSITÄT ZÜRICH

von

ETTORE DA PRA

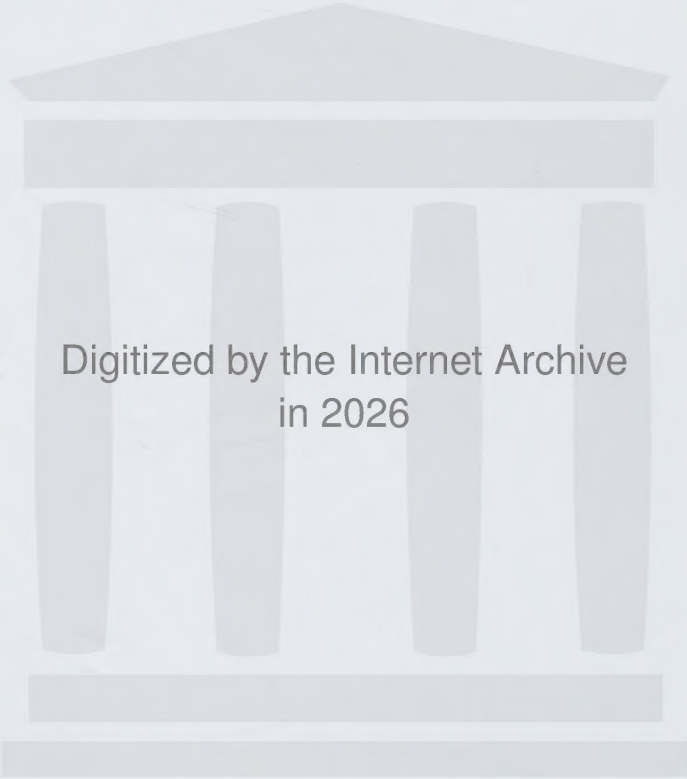
von Lozzo di Cadore (Italien)

Begutachtet von

Prof. Dr. R. Bachofen

Zürich 1987

Zentralstelle der Studentenschaft



Digitized by the Internet Archive
in 2026

Inhaltsverzeichnis

	Seite
1. Einleitung	2
2. Problemstellung	7
3. Material und Methoden	8
4. Resultate und Diskussion	20
4.1. Versuche mit Laboranlagen	20
4.1.1. Resultate mit dem vollständig durchmischten Fermenter	20
4.1.2. Resultate mit dem Anaerobfilter	23
4.1.3. Resultate mit der zweistufigen Anlage	28
4.1.4. Diskussion der Laborresultate	31
4.2. Betrieb der Pilotanlage in der Zuckerfabrik Frauenfeld	34
4.2.1. Kampagne 1981	34
4.2.2. Kampagne 1982	35
4.2.3. Kampagne 1983	38
4.2.4. Diskussion der Resultate mit der Pilotanlage	46
4.2.5. Energetische Aspekte für eine Grossanlage	47
4.3. Prozessoptimierung mit Hilfe eines Rechners	48
4.3.1. Die Struktur des Optimierungsprogrammes	48
4.3.2. Resultate mit dem Optimierungsprogramm	52
4.3.3. Diskussion der Optimierungsversuche	59
5. Zusammenfassung	60
6. Anhang	61
6.1. In der Arbeit verwendete Abkürzungen	61
6.2. Herleitungen	62
6.3. Programme	64
7. Literaturverzeichnis	93
8. Verdankungen	99
CURRICULUM VITAE	

1. Einleitung

Seit 1895 in der englischen Stadt Exeter die erste Biogasanlage zur Ausfäulung von kommunalem Klärschlamm in Betrieb genommen wurde [1], erreichte diese Technologie beim Kommunalabwassersektor einen Status, welcher heute den Betrieb grösserer Abwasserreinigungsanlagen nahezu ohne Einsatz von Fremdenergie erlaubt. Auf Grund niedriger Preise für Primärenergie, Problemen bei der Prozessstabilität und hoher Investitionskosten erlangte die Vergärung anderer organischer Materialien in industrialisierten Ländern bis heute keine grosse wirtschaftliche Bedeutung, obwohl sich die anaerobe Vorreinigung hochbelasteter Industrieabwässer als theoretisch beste Lösung anbietet. In einigen asiatischen Ländern hingegen haben Biogasanlagen einfacher Bauart, die zum Betrieb nur 4 GVE erfordern, eine weite Verbreitung gefunden. Etwa sieben Millionen solcher Anlagen sollen in China [2], über 70'000 in Indien in Betrieb sein [3].

Die Kenntnisse über die mikrobiellen Interaktionen bei der Vergärung organischer Materie beschränkten sich noch vor wenigen Jahren auf eine grobe Einteilung der daran beteiligten Mikroorganismengruppen und deren Reaktionen (Abbildung 1). Erst nachdem Fortschritte in der Isolation, Kultur und in der damit ermöglichten Charakterisierung der verschiedenen Methanogenen und der anderen an den Reaktionen beteiligten Bakterien erzielt worden waren, konnten an den Versuchsanlagen und an der Prozessführung gerichtete Änderungen vorgenommen werden [4].

Das langsame Wachstum der an der Vergärung beteiligten Methanogenen ist der Hauptgrund für Auswaschung, und verschiedene Methoden wurden zur Verminderung des Biomasseverlustes entwickelt, da die Dichte und Aktivität der Methanogenenpopulation in entscheidendem Masse die Leistungsfähigkeit des anaeroben Reinigungsverfahrens beeinflusst. Bedingt auch noch durch die Verschiedenartigkeit der zu behandelnden Abwässer entwickelte man funktionell unterschiedliche Verfahren, die aus energetischen Gründen weitgehend im mesophilen Bereich arbeiten und nachfolgend kurz charakterisiert werden sollen.

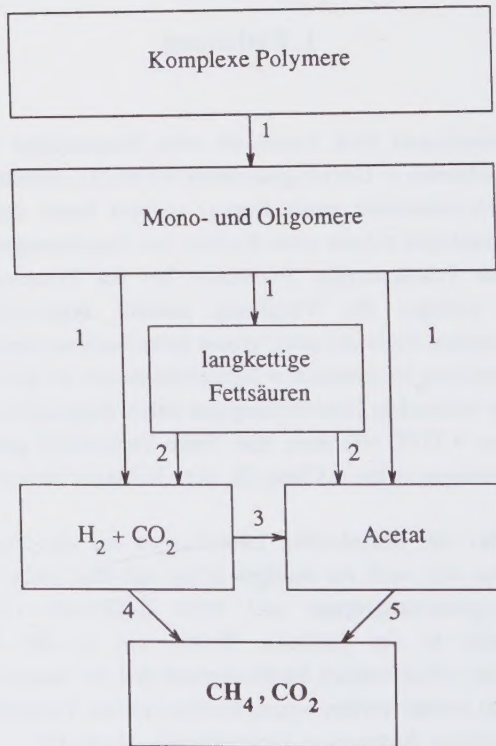


Abbildung 1: Beteiligte Mikroorganismengruppen im Kohlenstofffluss von polymeren Substanzen zu Methan und Kohlendioxyd nach Zinder [5].

Gruppe 1: Fermentative Bakterien

Gruppe 2: Wasserstoff produzierende acetogene Bakterien

Gruppe 3: Wasserstoff verbrauchende acetogene Bakterien

Gruppe 4: Kohlendioxyd reduzierende Methanogene

Gruppe 5: Acetoklastische Methanogene

Verfahren mit dem anaeroben Kontaktprozess

Dieses Verfahren wurde 1955 von *Schröpfer* entwickelt [6], und seither systematisch verbessert. Bei diesem Verfahren hängt die Reaktorleistung von der Sedimentationsfähigkeit der Mikroorganismen und der suspendierten Festteile im Sedimentationstank ab. Dieses Verfahren ist speziell für feststoffreiche Abwässer geeignet, da das System mit langen hydraulischen Retentionszeiten arbeitet und die Retentionszeit der Feststoffe durch ihre Rückführung aus dem Sedimentationstank in den Reaktor noch zusätzlich verlängert wird. Die spezifische Nettoenergieproduktion solcher Anlagen ist wegen ihrer Grösse und der langen hydraulischen Retentionszeiten von mindestens zehn Tagen sehr gering.

Verfahren mit dem Darmstadt-System (Plug-Flow Reaktor)

Plug-Flow oder Pfropfenströmung ist der laminare Durchfluss eines Substratpfropfens durch ein dünnes Rohr grosser Länge. Die üblicherweise errichteten kanal- bis wannenförmigen Reaktoren weisen ein Durchmesser/Länge-Verhältnis von 1:5 bis 1:10 auf. Eines der ersten Systeme dieser Art war die von *Reinhold und Noack* [7] entwickelte Gärkanalbiogasanlage System "Darmstadt". Ein langsam laufendes Rührwerk dient zur Durchmischung und Weiterförderung des Substrats. Dieses System findet ausschliesslich in der Landwirtschaft Anwendung. Der bei Masttierhaltungen anfallende Mist wird samt Einstreu dem Reaktor zugeführt. Der ausgegorene Festmist und die Jauche werden zur Düngung verwendet.

Verfahren mit dem Anaerobfilter

Dieses Verfahren wurde 1967 von *Young und McCarty* entwickelt [8]. Der Filter wird mit einem Trägermaterial gefüllt, das einerseits die Biomasse durch Bildung von Biofilmen im Reaktor zurückhält, andererseits die Gas-Flüssigkeit Trennung verbessert. Dieses Verfahren eignet sich vor allem zur Behandlung von Abwässern mit einem hohen Anteil an gelösten organischen Substanzen. Bei feststoffreichen Abwässern steigt die Gefahr einer Akkumulierung von Partikeln an, was zu einer Verstopfung der Anlage führen kann. Bei grösseren Anlagen besteht zudem noch das Problem der optimalen Flüssigkeitsverteilung am Filterboden. Durch eine ungeeignete Verteilung fliesst das Abwasser in wenigen Kanälen durch den Filter, was seine Reinigungsleistung bezogen auf das Filtervolumen sehr vermindert. Trotz zahlreicher Forschungsprojekte mit Anaerobfiltern sind heute nur wenige kommerzielle Anlagen dieses Typs in Betrieb. Das

Schwergewicht der Forschung auf diesem Gebiet liegt in der Entwicklung und Applikation leichter und die Bakterienadhäsion fördernder Trägermaterialien.

Verfahren mit dem upflow anaerobic sludge bed reactor (UASB)

Um der Gefahr einer möglichen Filterverstopfung entgegenwirken zu können, gelang es *Lettinga*, eine flockige, gut sedimentierbare Organismenpopulation zu gewinnen, die den Einsatz von Trägermaterialien zur Biomasserückhaltung erübrigte [9]. Bei diesem Verfahren gelang es auch, ein gutes System zur homogenen Verteilung des Substrates über den Filterquerschnitt zu entwickeln. Dieses Verfahren eignet sich wie der Anaerobfilter zur Behandlung von Abwässern mit einer hohen Konzentration an gelösten organischen Substanzen. Grossanlagen dieser Art werden seit Jahren in Holland, Belgien, Cuba und den USA betrieben.

Eine abgeänderte Form des UASB ist das Verfahren mit dem *fluidized and expanded bed reactor* [10]. Bei diesem Verfahren lässt man die Mikroorganismen auf gut sedimentierenden, inerten Trägerpartikeln wie Sand oder Aluminiumstaub anwachsen. Dieses Verfahren setzt jedoch eine hohe Durchflussrate voraus, da die Trägerpartikeln durch den Druck des aufwärtsfliessenden Abwassers angehoben und im Filter verteilt werden müssen, was die Betriebskosten stark ansteigen lässt und damit den Nettoenergiegewinn gegenüber den vergleichbaren Verfahren erheblich vermindert.

Verfahren mit dem downflow stationary fixed film reactor (DSFF)

Mit diesem von *van den Berg und Kennedy* entwickelten Filter wurde eine weitere Möglichkeit zur Verhinderung einer Filterverstopfung vorgestellt [11]. Der Filter enthält ein lamellenförmiges, ausgeklügelt angeordnetes Trägersystem, worauf die Mikroorganismen einen Biofilm bilden können. Losgelöste Biomasse und unlösliche Feststoffe werden mit dem Effluenten ausgespült. Die Installierung eines komplizierten Verteilersystems beim Zufluss fällt weg, da das eintretende Substrat durch die aufsteigenden Gasblasen homogen verteilt wird. Die organische Belastbarkeit des Filters ist einerseits limitiert durch die maximale Oberfläche mit Werten um $100 \text{ m}^2/\text{m}^3$, andererseits durch den maximalen Biofilmdurchmesser auf dem Trägermaterial, der seinerseits durch die Diffusion des Substrates limitiert wird. Dieses Verfahren wird in Puerto Rico und Kanada kommerziell angewandt.

Zweistufige Verfahren

Die Forderung nach Verfahrensstabilität auch bei organischen Schockbelastungen führte zudem noch zur Entwicklung von zweistufigen Verfahren, wobei Hydrolyse und saure Gärung von der Methangärung getrennt wurden. Von dieser Methode erhofft man sich eine Verbesserung der Durchsatzleistung bei der Vergärung organisch hochbelasteter Abwässer, da die Umsetzung von Kohlenhydraten zu Fettsäuren, je nach gewünschter Fettsäurezusammensetzung, im pH-Bereich von 4-6 stattfindet [12], bei eiweissreichen Substraten pH-Werte zwischen 6 und 7 für die Hydrolyse günstig sind [13], die beiden letzten Schritte der Acetat- und Methanbildung hingegen im schwach sauren bis alkalischen Bereich mit pH-Werten zwischen 6.5 und 8 ablaufen.

Durch den Einsatz von zweistufigen Anlagen konnten wie erwartet die biologische Umsatzrate und die Methanausbeute erhöht werden. Der konstruktive Mehraufwand und die erhöhte Komplexität der Prozessführung mit zusätzlichen Regelkreisläufen verhindern jedoch noch heute eine häufigere Erstellung von Grossanlagen dieser Art.

2. Problemstellung

Bei der Zuckerrübenverarbeitung in der Zuckerfabrik Frauenfeld fallen täglich über 100 m³ organisch hochbelastetes Schwemmwasser mit COD-Werten bis zu 30 g/l an. Dieses Abwasser kann nicht direkt der kommunalen Kläranlage zugeführt werden, da ihre biologische Stufe sehr empfindlich ist gegenüber hohen Konzentrationen an organischem Material und die Gefahr eines Ausfalls dieser Stufe besteht. Das Schwemmwasser fließt deshalb in ein Speicherbecken, worin die organischen Substanzen über eine Zeitspanne von mehreren Monaten vergärt werden und eine spätere Endreinigung des Abwassers durch dosiertes Einleiten in die Kläranlage möglich wird. Da bei der Vergärung von Rübenschwemmwasser unter anderem Methan entsteht, könnte bei einer anaeroben Vorreinigung des Schwemmwassers bis zu 5% des gesamten Energiebedarfs der Zuckerfabrik durch die Verwertung des Methans gewonnen und das behandelte Schwemmwasser direkt einer Kläranlage zugeführt werden.

Ziel dieser Arbeit war es, mehrere heute angewendete Methoden zur anaeroben Behandlung von Abwässern in Bezug auf ihre Eignung zur Vorreinigung von Rübenschwemmwasser im Labormassstab miteinander zu vergleichen.

In einer zweiten Phase sollte der Betrieb einer Pilotanlage in der Zuckerfabrik Aufschlüsse über die geometrische Auslegung, die Handhabung der Füllung, die Anfälligkeit gegenüber Verstopfungen und die Prozessführung unter realistischen Bedingungen geben.

Die rechnerunterstützte Erfassung, Speicherung und Auswertung von Daten biologischer Prozesse ist heute zum Standardverfahren geworden. Die Prozessoptimierung steckt hingegen noch in ihren Anfängen und konzentriert sich vor allem auf einfache Systeme. Zur Optimierung der Abbauleistung der Laboranlage sollte in einer dritten Phase aus den früher erhaltenen Betriebsdaten Kenndaten ermittelt werden, welche einen Operator simulieren. Die Rechnersimulation sollte eine automatische Regulierung der Zuflussrate des Schwemmwassers in den Filter in Abhängigkeit der COD-Konzentration des Schwemmwassers ermöglichen.

3. Material und Methoden

Rübenschwemmwasser als Substrat

Das Schwemmwasser wurde von der Zuckerfabrik Frauenfeld bezogen. Für die Laborversuche bewahrte man es in Flaschen im Kühlraum auf, die Pilotanlage hingegen betrieb man direkt mit Schwemmwasser aus dem Fabrikkreislauf.

Tabelle 1 gibt die Elementarzusammensetzung des Schwemmwassers an. Die Analyse wurde an der EAF Birmensdorf mit einem ICP-AES (Inductive Coupled Plasma - Atomic Emmission Spectrometer) durchgeführt.

Die Schwemmwasseranalyse der Kampagne 1983 ergab einen Gehalt an Gesamtphosphat von 23.7 mg/l, an Gesamtstickstoff von 109.4 mg/l. Der Kohlenstoffgehalt betrug 4620 mg/l bei einem COD-Wert von 7580 mg/l.

<i>Element</i>	<i>Gehalt mg/l</i>	<i>Element</i>	<i>Gehalt mg/l</i>
Al	3.62	Mg	28.65
B	0.137	Mn	4.547
Ca	335.88	Mo	0.001
Cd	0	Na	510.95
Co	0.006	Ni	0.058
Cr	0.049	P	65.65
Cu	0.043	Pb	0.18
Fe	103.17	S	7.37
Hg	0.031	Si	8.94
K	64.64	Sn	0.08
Li	0	Zn	0.502

Tabelle 1: Elementaranalyse des Schwemmwassers (Dezember 1984)

Impfmateriail

Als Inokulum diente Faulschlamm aus der Kläranlage Zumikon. In den Laborversuchen bewirkte eine Zentrifugation (ca. 3500 U/min, 12 min, Zentrifuge UJ3, M. Christ, Osterade, BRD) eine gute Trennung der Biomasse (grün-schwarze Schicht) von den sandigen Komponenten des Schlammes. Das so gewonnene Impfmateriail wurde anschliessend im Schwemmwasser (ca. 46g TS/l) resuspendiert und während drei Wochen in einem Fermenter an das Schwemmwasser adaptiert (37 °C, 300 U/min, pH 7.2).

Zum Aufstarten der Pilotanlage in Frauenfeld benützte man im Jahr 1981 80 l unbehandelten Klärschlamm. Im Jahre 1983 bestand das Inokulum aus 20 l unbehandeltem Klärschlamm, zusätzlich wurden 8 l adaptierter Klärschlamm aus dem Laborfermenter zugegeben.

Trägermaterial

Die Füllung des Anaerobfilters im Labor bestand aus Blähtonkugeln von 5-10mm Durchmesser, diejenige der Pilotanlage aus solchen von 10-20mm Durchmesser.

Beim Füllen der Pilotanlage war besonders darauf zu achten, dass möglichst wenige Blähtonkugeln beschädigt wurden, da diese Bruchstücke zu Verstopfungen in den Zu- und Ableitungen führen konnten. Der Filter wurde dazu zuerst ganz mit Wasser gefüllt und die Blähtonkugeln dann vorsichtig auf die Wasseroberfläche aufgeschichtet. Die Wassersäule wurde dann langsam gesenkt, bis der ganze Filter mit Blähtonkugeln gefüllt war. Nach einer gründlichen Reinigung der Füllung mit Wasser wurde der Filter mit Schwemmwasser und Klärschlamm von oben gefüllt und anschliessend mit einer Vakuumpumpe während 15 Stunden entgast.

Gasanalysen

Manuelle Gasanalysen an einzelnen Proben erfolgten mit einem Carle 111 Analytical Gas Chromatograph, ausgerüstet mit einem Wärmeleitfähigkeitsdetektor (Carle, Anaheim CA, USA), gekoppelt an einen Integrator (Spectra-Physics, Santa Clara CA, USA). Die Auftrennung erfolgte auf einer MS5A, bzw. CSS Säule (Supelco Inc., Bellefonte PA, USA).

Automatische Gasanalysen erfolgten mit einem GOW-MAC 69-550 Gaschromatographen (GOW-MAC Instruments CO, Madison NJ, USA), ausgerüstet mit einem pneumatischen Loop (Carlo Erba S.P.A., Milano, I). Der Loop wurde mit einem Timer angesteuert. Zur Auftrennung benutzte man eine Carbosieve B 120/140 Säule.

Die kontinuierliche Bestimmung des Methangehaltes erfolgte mit einem Horiba PIR 2000 Infrarot Gasanalysator (Horiba Instruments SA, Chavannes-de-Bogis).

Bestimmung des Gasflusses

Zur Messung der Gasaustrittsrate verwendete man bis 1983 für die Laborversuche und bei der Pilotanlage LC1U Gaszähler (H. Wohlgroth & Co, Zürich), ausgerüstet mit einem opto-elektronischen Signalwandler. Für die Optimierungsversuche mit dem Anaerobfilter im Labor wurden Tylan Masseflussmeter eingesetzt (Tylan, Carson CA, USA).

Analyse der flüchtigen Fettsäuren

1.0 ml der Proben aus dem Anaerobfilter wurden mit 50 µl 1N NaOH versetzt und bis zur Trockne eingedampft, der Rückstand in 0.5 ml 50% Ameisensäure aufgenommen und 1 µl davon analysiert. Die Bestimmung erfolgte mit einem Gaschromatographen Shimadzu GC-7 (Shimadzu Corp., Kyoto, Japan). Die Säule war gefüllt mit 60/80 Carbopac C, 0.3% Carbowax 20M, 0.1% Phosphorsäure (Supelco Inc., Bellefonte PA, USA). Als Trägergas diente mit Ameisensäuredampf gesättigtes Helium [15].

Reinigung der verwendeten Gase

Die Reinigung des Stickstoffs 45 (Carbagas AG, Zürich mit $[O_2] < 5 \text{ ppm}$) zu einem Gas mit $< 0.1 \text{ ppm}$ Restsauerstoff erfolgte mit einem nach Dubach [16] modifizierten Hungate-Ofen [17].

Labor-Versuchsanlagen

Alle Geräte ohne Herkunftsangaben sind Eigenentwicklungen des Institutes für Pflanzenbiologie der Universität Zürich.

Vollständig durchmischte Fermenter

Der verwendete Fermenter, sowie das zugehörige Mess- und Regelsystem wurden bereits mehrfach detailliert beschrieben [18, 19]. Abbildung 2 zeigt die Versuchsanordnung. Das Füllvolumen im Fermenter wurde durch die Waage und die Pumpe 2 konstant gehalten. Das Sedimentationsgefäß bestand aus einem Schütteltrichter (1 l), die Rückführung der darin sedimentierten Biomasse in den Fermenter erfolgte periodisch mit Pumpe 3. Die Gasproduktion wurde kontinuierlich gemessen und auf einem Mehrpunktschreiber PM8235 aufgezeichnet (Philips, Eindhoven, NL), die Konzentration der in der Probe vorkommenden Gase extern am Carle 111 bestimmt. Die Probe zur Analyse des behandelten Schwemmwassers wurde vor dem Sedimentationsgefäß nach der Pumpe 2 entnommen. Die Betriebstemperatur im Fermenter betrug 35°C bei einem pH-Wert von 7.2 und einer Rührgeschwindigkeit von 300 U/Minute. Diese Betriebsbedingungen wurden von *Gosh* übernommen [20].

Anaerobfilter im Labormassstab

Der von *W. Tesch* konstruierte Filter [21, 22] wurde leicht abgeändert:

Die Thermostatisierung erfolgte neu mit einem Wasserbad, da die Heizpatronen bei schwankender Wasserzufuhr oft durchbrannten. Die pH-Regelungseinrichtung wurde entfernt. Die Filtertemperatur war 35 °C gemäss *Ramalho* [23].

Abbildung 3 zeigt die Versuchsanordnung. Filtertemperatur und Gasbildung wurden kontinuierlich gemessen und mit dem Philips Schreiber aufgezeichnet. Die Gasanalyse erfolgte periodisch mit dem GOW-MAC alle 60 Minuten.

Zweistufige Versuchsanlage

Die erste Stufe bestand aus dem oben beschriebenen Fermenter in der Funktion eines BCCV (Biological Conditioning & Control Vessel), die zweite Stufe aus dem Anaerobfilter. Abbildung 4 zeigt die Versuchsanordnung. Die Pumpe 2 führte dem Anaerobfilter kontinuierlich mit einer vorgegebenen Rate Substrat 2 aus dem Fermenter zu. Die Pumpe 1 und die Waage dienten der Regulierung des Flüssigkeitsgewichts im

Fermenter. Die Gasbildung des Fermenters wurde periodisch, die des Filters, sowie die Temperatur des Filters wurden kontinuierlich gemessen und mit dem Philips Schreiber aufgezeichnet. Für beide Einheiten diente der Carle 111 zur gaschromatographischen Analyse. Die Bestimmung des Schwemmwasser-pH am Überlauf erfolgte periodisch. Die Temperatur im Fermenter betrug 30°C, der pH-Wert 6.0 und die Drehzahl 300 U/Minute. Die Filtertemperatur betrug 35 °C. Die Betriebsbedingungen der zweistufigen Versuchsanordnung wurden von *Zoetemeyer* übernommen [24].

Die Pilotanlage in der Zuckerfabrik Frauenfeld (ZFF)

Die Anlage ist in Abbildung 5 dargestellt.

Zum Bau des Filters wurde ein Metallrohr von 48 cm Durchmesser und 380 cm Höhe verwendet. Sieben Schaugläser (12 cm Durchmesser) ermöglichten die Beobachtung und einen eventuellen Eingriff in das Innere des Filters. Die Anschlüsse für den Zulauf bestanden aus zwei 1/2 Zoll-Stutzen am unteren konischen Ende des Filters. 35 cm oberhalb der Stutzen war eine Lochplatte angebracht. Sie sollte eine homogene Verteilung des zugepumpten Substrates gewährleisten und eventuelle Bruchstücke des Füllmaterials von den Zulaufschläuchen fernhalten. Am oberen Ende des Filters wurden zwei Abflusstutzen angebracht (1/2 Zoll Durchmesser). Den Abschluss bildete ein gasdicht verschraubter Deckel mit einem zentralen Stutzen für den Gasaustritt. Seitlich am Reaktor waren in axialer Richtung und äquidistant acht radial verschiebbare Probeentnahmerohre angebracht. Das Temperaturprofil im Filter wurde von sieben Pt100 Fühlern längs der Achse gemessen.

Das gebildete Gas strömte aus Deckel- und oberem Abflusstutzen über einen Rückflusskühler zur Gasuhr. Die Schwemmwasserzuführung erfolgte aus dem geschlossenen Kreislauf der ZFF über zwei Filter (2 mm, bzw. 0.5 mm Porendurchmesser) und über ein Ausgleichsgefäß von 40 l Volumen mit Schwimmerventil. Eine Watson-Marlow 601 Schlauchquetschpumpe (Skan AG, Basel) pumpte das Schwemmwasser aus dem Ausgleichsgefäß in den Reaktor. Nach der Pumpe wurde im Zufuhrschlauch ein Rückschlagventil angebracht, um bei etwaigen Schlauchdefekten ein Auslaufen des Reaktorinhalts zu verhindern.

Der Filter wurde nicht thermostatisiert, da die Raumtemperatur am Standort des Filters in der Fabrikhalle immer zwischen 30 °C und 37 °C betrug.

Datenerfassung und Probenaufbereitung:

Temperaturprofil, Raumtemperatur und Gasbildungsrate wurden von einem 12-Punkte Schreiber kontinuierlich aufgezeichnet. Die COD-Bestimmungen für Zufluss (aus dem Ausgleichsgefäß), wie auch für das behandelte Schwemmwasser, wurden durch das Labor der ZFF alle zwei Tage vorgenommen. Der pH des Schwemmwassers am Überlauf wurde alle Tage gemessen. Die Analyse der Anaerobfilterprofile (COD-Werte und Fettsäurenkonzentrationen), sowie die Bestimmung der Gaszusammensetzung erfolgten im Laboratorium am Institut für Pflanzenbiologie. Für die Gasprobenentnahme wurde am Austritt der Gasuhr ein Nadelaufsatz angebracht. Er ermöglichte, Hungate-Röhrchen mit Gas zu füllen.

Labor-Versuchsanlage zur Optimierung der Filterbelastung

Die Versuchsanordnung ist in Abbildung 6 dargestellt.

Anaerobfilter:

Die schon zuvor beschriebene Anlage wurde wie folgt abgeändert und ergänzt. Zur kontinuierlichen Erfassung der pH- und Redoxwerte wurde eine kombinierte Elektrode (Pt 405-60-85, Dr. W. Ingold AG, Urdorf) durch einen Stutzen im Deckel des Filters in die Lösung eingeführt. Die Abdichtung erfolgte durch einen O-Ring. Um exakte Gasflusswerte zu erhalten, trocknete man das Gas zusätzlich mit Kalziumchlorid und Silica-Gel. Das Sulfid im Abgas entfernte man durch Absorption mit Eisenhydroxid. Die Gasflussbestimmung erfolgte mit einem Tylan Masseflussmeter, dem als Schutz gegen Flüssigkeitseintritt ein hydrophober Membranfilter Gelman ACRO 50 (Skan AG, Basel) vorgeschaltet war. Der Methangehalt wurde mittels eines Infrarot-Gasanalysators kontinuierlich bestimmt, oder, falls das Gerät eine zweite Gasquelle zu messen hatte, über eine Umschaltanlage in frei wählbaren Intervallen. Die Methanbildungsrate wurde mit dem Rechner aus dem Gasfluss und der Gaszusammensetzung ermittelt.

Rechnerhardware:

Die Rechneinheit setzte sich aus VME-Bus kompatiblen Prozessor- und I/O-Karten zusammen.

- VEX-111 CPU-Karte (Eltec, Mainz, BRD) mit dem 68000 Mikroprozessor von Motorola, zwei Megabyte lokales Memory und acht seriellen Schnittstellen RS-232.
- D/A-Interface MVP 904: 16 Kanäle, 0-10 Volt, 12 bit Auflösung (Burr-Brown Ltd., Livingston, GB).
- A/D-Interface MVP 901: 32 Kanäle single ended, 12 bit Auflösung (Burr-Brown Ltd.).
- Relais-Interface EFD-R320: 32 Relaisausgänge. (Thompson Semiconducteurs, Velizy, F).
- Farbgraphik-Karte Graz3 (Eltec) mit eigenem Graphikprozessor zur Entlastung des Hauptprozessors. 512x256 Punkte Auflösung mit RGB- und Sync. Ausgängen, sowie Anschluss für ein Lightpen.

Betriebssystem:

Es wurde das OS9-68000 real time, multi tasking, multi user Betriebssystem eingesetzt (Microware, Des Moines Ill, USA).

Software:

Verwendet wurde das PCS-68000 Programm (entwickelt von K. Schneider, Institut für Pflanzenbiologie, Universität Zürich) zur Datenerfassung und Auswertung. Mit den dazugehörenden Interfacefunktionen war es möglich, auf die internen Datenstrukturen des PCS-68000 zuzugreifen. Die Programmiersprache war "C" [25].

Signalverarbeitung:

Die Signale zwischen den verschiedenen Messstationen und dem Rechner wurden mit Trennverstärkern galvanisch getrennt.

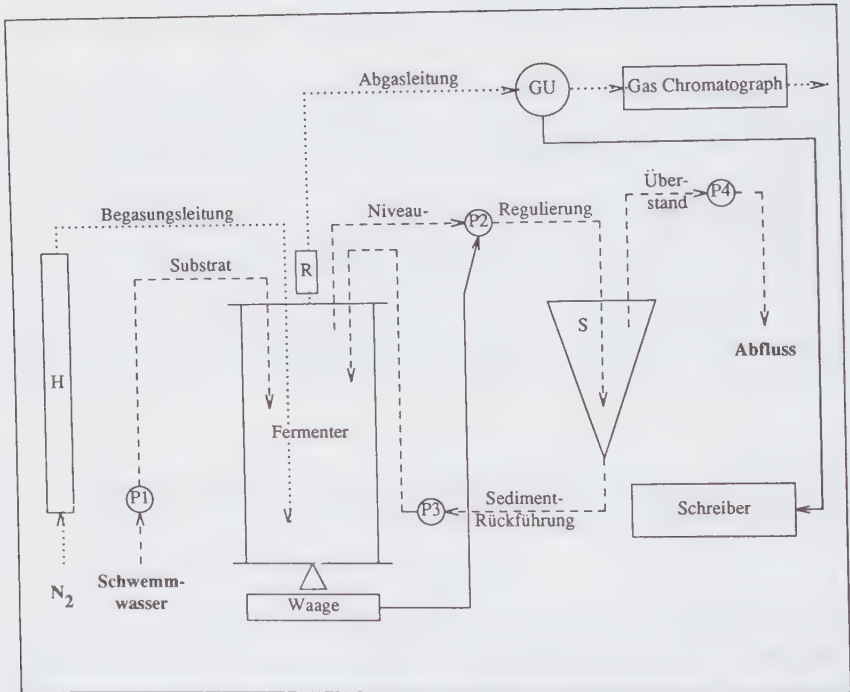


Abbildung 2: Schematische Darstellung der Versuchsanlage mit dem vollständig durchmischten Fermenter.

Abkürzungen:

GU: Gasuhr; H: Hungate Ofen; P1: Substratpumpe, konstanter Betrieb; P2: Niveauregulierungspumpe, gesteuert durch die Waage; P3: Sediment-Rückführungspumpe, gesteuert durch ein Impulsgerät; P4: Abflusspumpe, konstanter Betrieb; R: Rückflusskühler; S: Sedimentationsgefäß;

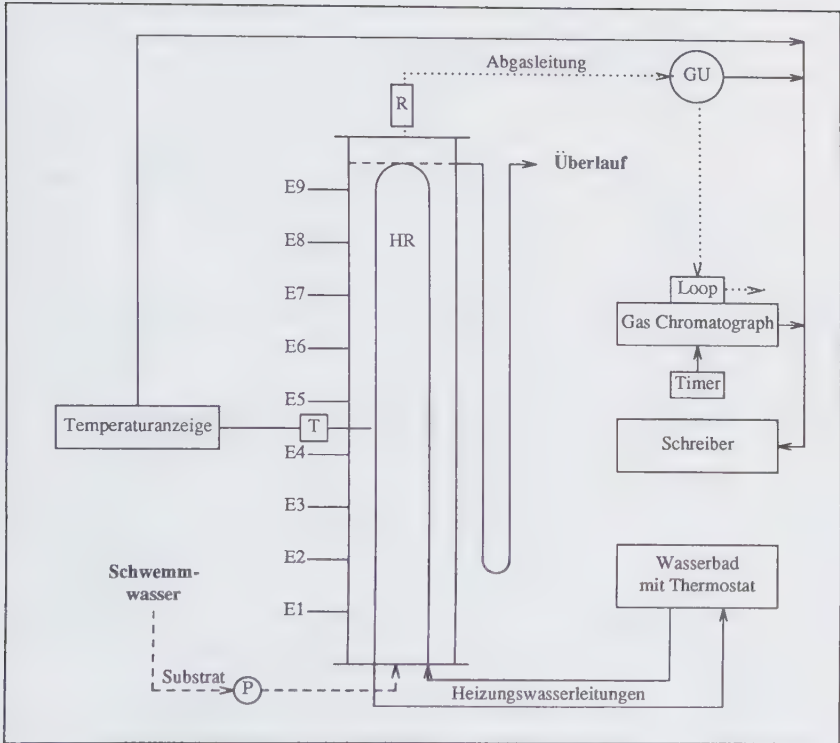


Abbildung 3: Schematische Darstellung der Labor-Versuchsanlage mit dem Anaerobfilter.

Zeichenerklärung:

E1 .. E9: Probeentnahmestellen; **GU:** Gasuhr; **HR:** Heizrohr; **P:** Substratpumpe, konstanter Betrieb; **R:** Rückflusskühler; **T:** Temperatursonde (Pt100);

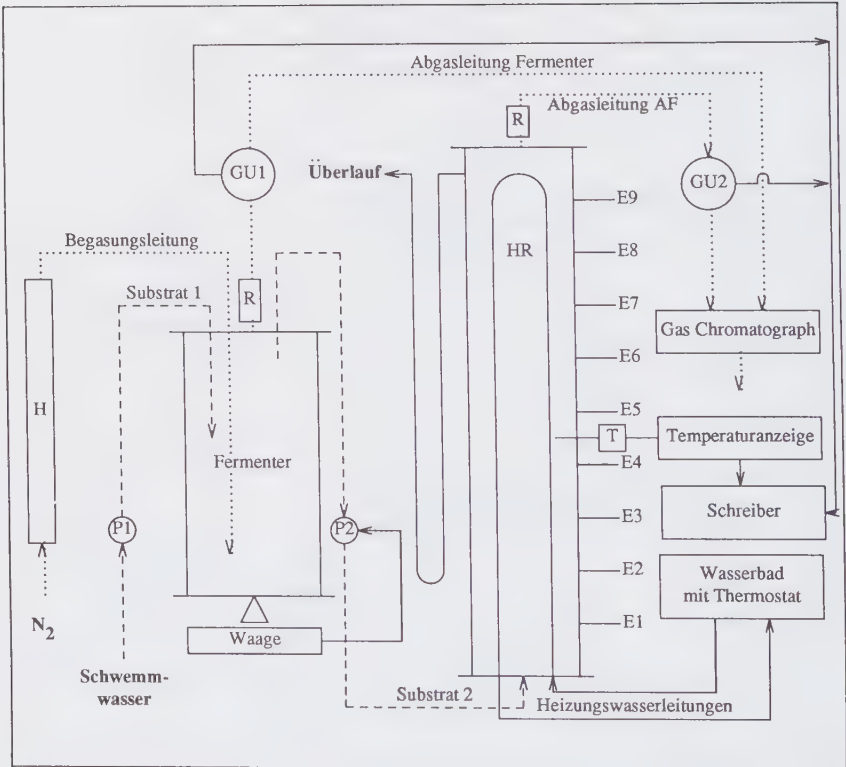


Abbildung 4: Schematische Darstellung der zweistufigen Labor- Versuchsanlage.

Zeichenerklärung:

E1 .. E9: Probeentnahmestellen; **GU1:** Gasuhr für Fermenterabgas; **GU2:** Gasuhr für Anaerobfilterabgas; **HR:** Heizrohr; **H:** Hungate Ofen; **P1:** Niveauregulierungspumpe, gesteuert durch die Waage; **P2:** Substratpumpe für den Anaerobfilter, konstanter Betrieb; **R:** Rückflusskühler; **T:** Temperatursonde (Pt100);

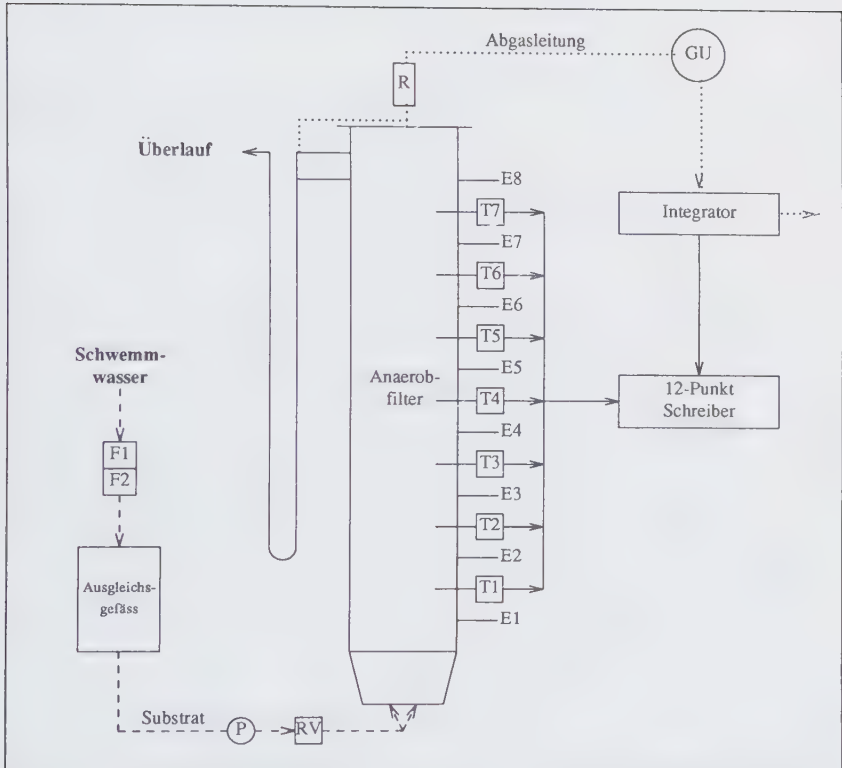


Abbildung 5: Schematische Darstellung der 680 l Pilotanlage in der Zuckerfabrik Frauenfeld.

Zeichenerklärung:

E1 .. E8: Probeentnahmestellen; **F1,F2:** Filter; **GU:** Gasuhr; **P:** Substratpumpe; **R:** Rückflusskühler; **RV:** Rückschlagventil; **T1 .. T7:** Temperatursonden (Pt100);

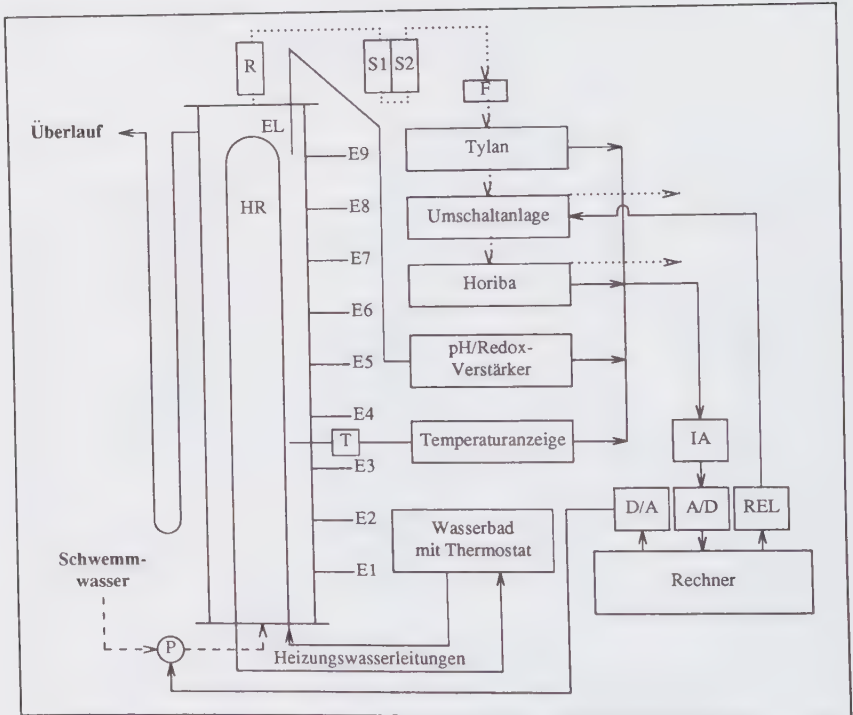


Abbildung 6: Schematische Darstellung der Labor-Versuchsanlage mit der Erfassung der on-line Parameter für die Optimierung der Filterbelastung.

Zeichenerklärung:

A/D: Analog/Digital Wandler; D/A: Digital/Analog Wandler; E1 .. E9: Probeentnahmestellen; EL: Kombinierte pH-/Redox Elektrode; F: Hydrophober Membranfilter; HR: Heizrohr; IA: Trennverstärker; P: Substratpumpe, angesteuert vom Rechner; R: Rückflusskühler; REL: Relaiskarte; S1: Trockenrohr mit $\text{Fe}_2\text{O}_3 \cdot n\text{H}_2\text{O}$; S2: Trockenrohr mit CaCl_2 und Silica Gel; T: Temperatursonde (Pt100);

4. Resultate und Diskussion

4.1 Versuche zur Behandlung von Rübenschwemmwasser im Labormassstab

4.1.1. Resultate mit dem vollständig durchmischten Fermenter

Während des Anlaufprozesses im Fermenter (Aufbau in Abbildung 2) konnte man während 17 Tagen H_2 und CO_2 im gebildeten Gas messen. Der Wasserstoff nahm jedoch mit der einsetzenden Methanbildung ab und konnte nach den 17 Tagen nicht weiter nachgewiesen werden (Abbildung 7). Während des anschliessenden, 70 Tage dauernden Versuches erhöhte man die Belastung im Fermenter mit gleichzeitig abnehmender hydraulischer Retentionszeit von $1.48 \text{ kg COD} \cdot \text{m}^{-3} \cdot \text{d}^{-1}$ bis auf $11.4 \text{ kg COD} \cdot \text{m}^{-3} \cdot \text{d}^{-1}$ (Abbildung 7). Die Verdünnungsrate wurde jeweils erst erhöht, nachdem die spezifische Methanproduktion P während 20 Stunden konstant geblieben war. Nach jeder Verkürzung der Verweilzeit konnte für kurze Zeit eine reduzierte Gasproduktion beobachtet werden. Die spezifische Gasproduktionsrate erreichte mit $1.61 \text{ m}^3 \text{CH}_4 \cdot \text{m}^{-3} \cdot \text{d}^{-1}$ ein Maximum bei einer Belastung von $3.56 \text{ kg COD} \cdot \text{m}^{-3} \cdot \text{d}^{-1}$ und einer Verweilzeit von 57 Stunden.

Eine pH-Regelung war bei einer Retentionszeit von 120 Stunden nicht notwendig, es stellte sich im Fermenter selbstständig ein stabiler pH-Wert um 7.1 ein. Bei den kürzeren Retentionszeiten begann das System zu übersäuern und der pH-Wert musste kontinuierlich geregelt werden. Bei einer Retentionszeit von 43 Stunden sank die spezifische Methanproduktionsrate gegen null; bei einer Retentionszeit von 17 Stunden konnte im gebildeten Gas kein Methan mehr nachgewiesen werden. Die Erhöhung der Verdünnungsrate bewirkte auch eine Verminderung der prozentualen COD-Abbauleistung COD_I von anfänglich 84% auf nur noch 4% bei 17 Stunden Verweilzeit, ebenso ging die Methankonzentration im gebildeten Gas von 76% auf 0% zurück (Tabelle 2).

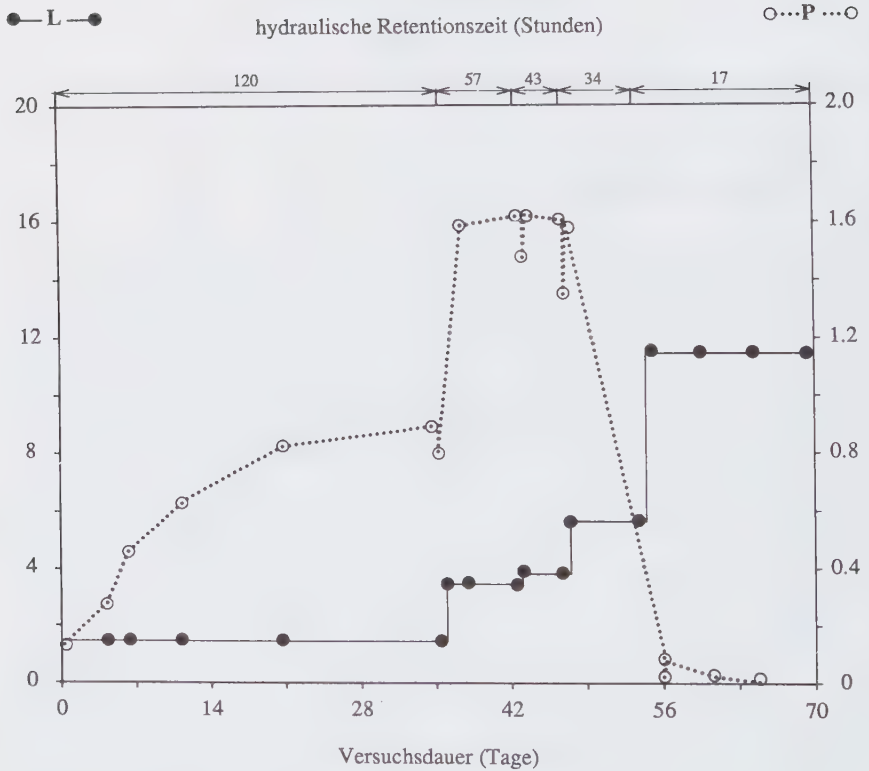


Abbildung 7: Zeitlicher Verlauf von Raumbelastung L ($\text{kg COD} \cdot \text{m}^{-3} \cdot \text{d}^{-1}$) und spezifischer Methanproduktionsrate P ($\text{m}^3 \text{CH}_4 \cdot \text{m}^{-3} \cdot \text{d}^{-1}$) als Funktion der Versuchszeit und der hydraulischen Retentionszeit als Parameter.

Mit der Verkürzung der Retentionszeit veränderten sich die Fettsäurenkonzentrationen im Fermenter. Bei einer Retentionszeit von 43 Stunden konnte erstmals Butyrat im Fermenterüberlauf festgestellt werden, bei 17 Stunden auch Isobutyrat. Die Konzentrationen der übrigen Fettsäuren im Fermenter stiegen ebenfalls an und erreichten bei der kürzesten Verweilzeit beinahe die Konzentrationen der Fettsäuren im zugeführten Substrat (Tabelle 3).

Als besonderes Problem erwies sich die Rückführung der produzierten Biomasse in den Fermenter. Die sich im Sedimentationstrichter bildenden Gasblasen verhinderten ein ungestörtes Absetzen und damit eine effiziente Biomasserückführung. Betroffen war vor allem die langsamwachsende Gruppe der Methanogenen, so dass bei einer Verweilzeit von 5 Stunden erneut H_2 im Gas auftrat. Die COD-Reduktion betrug dabei immer noch 8%, erst bei einer Verweilzeit unter 3 Stunden begann das Auswaschen der acetogenen Bakterien.

Verweilzeit Stunden	L $kg\ COD \cdot m^{-3} \cdot d^{-1}$	$[CH_4]$ %	P $m^3 CH_4 \cdot m^{-3} \cdot d^{-1}$	COD_r %
120	1.48	76	0.78	84
57	3.56	72	1.61	78
43	3.96	64	1.59	74
34	5.72	18	0.03	9
17	11.40	0	0	8
5	32.15	0	0	8
3	58.20	0	0	2

Tabelle 2: Betriebsdaten des Fermenters. Dargestellt sind die organische Filterbelastung **L**, die Methankonzentration $[CH_4]$ im gebildeten Gas, die spezifische Methanbildungsrate **P** und der prozentuale COD-Abbau COD_r in Abhängigkeit der **Verweilzeit**.

Verweilzeit Stunden	Probe	Acetat mM	Propionat mM	Butyrat mM	Isobutyrat mM
120	Zulauf	92.0	31.0	2.0	0.2
	Überlauf	0.7	0.2	0	0
57	Zulauf	119.0	38.0	7.0	0.6
	Überlauf	1.4	0.5	0	0
43	Zulauf	94.0	29.0	2.0	0.3
	Überlauf	1.7	0.8	0.1	0
34	Zulauf	114.0	36.0	5.0	0.6
	Überlauf	67.0	17.0	2.8	0.1
17	Zulauf	116.0	38.0	6.0	0.8
	Überlauf	102.2	27.6	3.2	0.3

Tabelle 3: Fettsäurekonzentrationen im Substrat (=Zulauf) und im Fermenterüberlauf.

4.1.2. Resultate mit dem Anaerobfilter

Impfmateriel für die Inbetriebnahme des Filters war 500 ml Klärschlamm, adaptiert in einem Laborfermenter (300 U/min, 35°C, pH 7.0), der kontinuierlich zur Bereitstellung von Animpfmateriel betrieben wurde. 9 Tage nach dem Animpfen erlaubte eine steigende Methanbildungsrate den Beginn der Versuche. Während der stufenweisen Erhöhung der Raumbelastung bis zu $8.85 \text{ kg COD} \cdot \text{m}^{-3} \cdot \text{d}^{-1}$ stellte sich im Anaerobfilter selbstständig ein stabiler pH-Wert um 7.1 ein.

Nach jeder Verringerung der hydraulischen Retentionszeit konnte kurzfristig eine pH-Senkung im Filterüberlauf gemessen werden (Pfeile in Abbildung 8). Der pH-Wert am Überlauf sank bei der kürzesten Verweilzeit von 22 Stunden und damit maximalen Filterbelastung stetig und betrug bei Versuchsabbruch 6.95 (Abbildung 8).

Die spezifische Methanproduktionsrate stieg stetig mit der Filterbelastung und erreichte für kurze Zeit den höchsten Wert von $4.05 \text{ m}^3 \text{CH}_4 \cdot \text{m}^{-3} \cdot \text{d}^{-1}$ bei einer Verweilzeit von 22 Stunden. Die Methanproduktionsrate blieb jedoch nach Erreichen des Maximalwertes nicht stabil sondern begann, ähnlich dem pH-Wert, stetig zu sinken, was eine Überbelastung des Filters andeutete. Um eine Übersäuerung und somit eine lange Erholungszeit für den Filter zu vermeiden, wurde die Substratzufuhr gestoppt.

Tabelle 4 zeigt die prozentuale Abbauleistung COD_r in Abhängigkeit der Raumbelastung L und der hydraulischen Retentionszeit. Erst bei einer Belastung von über $4 \text{ kg COD} \cdot \text{m}^{-3} \cdot \text{d}^{-1}$ sank die Abbauleistung deutlich unter 80% (Tabelle 4).

Um den Einfluss von hydraulischer Retentionszeit, getrennt von der Raumbelastung auf die Abbauleistung des Anaerobfilters untersuchen zu können, wurde Schwemmwasser in einem Rotationsverdampfer aufkonzentriert. Dies ermöglichte es, den Filter bei einer Verweilzeit von 45 Stunden mit einer Raumbelastung von $8.4 \text{ kg COD} \cdot \text{m}^{-3} \cdot \text{d}^{-1}$ zu betreiben (bei den vorangegangenen Versuchen hatte die Retentionszeit 22 Stunden und die Belastung $8.85 \text{ kg COD} \cdot \text{m}^{-3} \cdot \text{d}^{-1}$ betragen). Die Verdoppelung der Retentionszeit bei gleichbleibender Belastung bewirkte eine Erhöhung der Methanproduktionsrate, sowie eine Verbesserung der Abbauleistung (Tabelle 4).

Tabelle 5 zeigt zusammenfassend die Fettsäurekonzentrationen im Substrat und im Filterüberlauf. Wie schon aus den pH-Werten des Überlaufs vermutet werden konnte, waren die Fettsäurekonzentrationen im behandelten Schwemmwasser über alle Belastungsstufen hinweg sehr gering; erst bei einer Retentionszeit von 22 Stunden tauchten Spuren von Butyrat im Ausfluss auf. Isobutytrat trat nie auf.

Zur Bestimmung der effektiven Retentionszeit im Filter wurde bei mehreren Durchflussraten Fluoresceinlösung als Tracer in den Filter gepumpt und die Zeit bis zur Detektion am Überlauf gemessen. Abbildung 9 zeigt, dass die gemessenen Retentionszeiten sich erheblich von den theoretischen Werten unterschieden, dies vor allem bei den kleineren Verdünnungsraten.

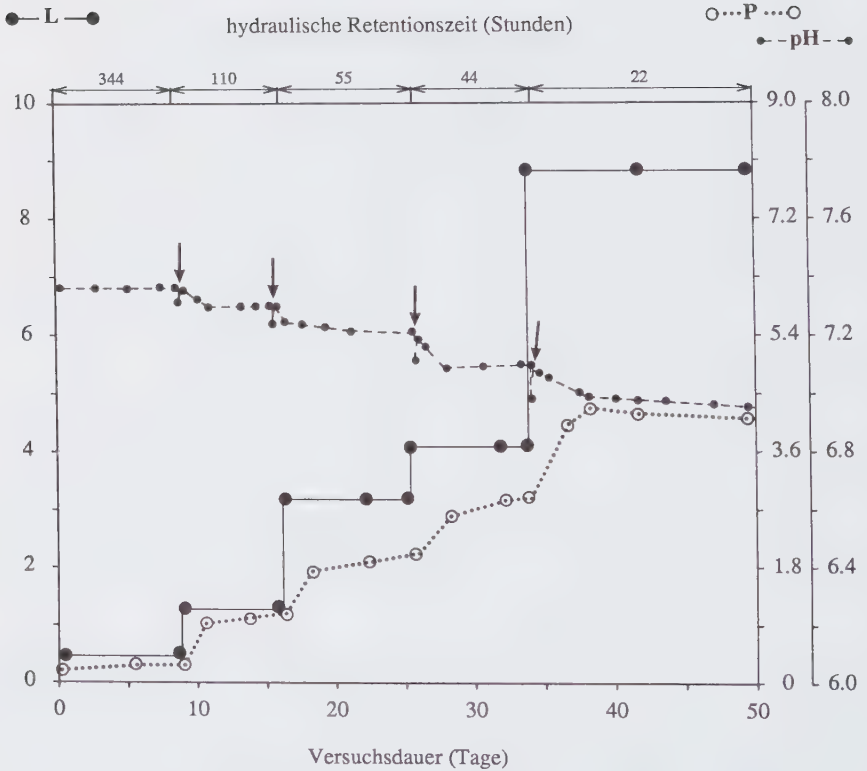


Abbildung 8: Zeitlicher Verlauf von Raumbelastung L (kg COD $\cdot$ m $^{-3}$ $\cdot$ d $^{-1}$), spezifischer Methanproduktionsrate P (m 3 CH $_4$ $\cdot$ m $^{-3}$ $\cdot$ d $^{-1}$) sowie des pH-Wertes als Funktion der Versuchszeit und der hydraulischen Retentionszeit als Parameter.

Verweilzeit Stunden	L kg COD*m ⁻³ *d ⁻¹	[CH ₄] %	P m ³ CH ₄ *m ⁻³ *d ⁻¹	COD _r %
344	0.49	82	0.25	91
110	1.29	77	0.85	88
55	3.19	75	2.01	86
44	4.17	74	2.79	80
22	8.85	70	4.05	68
45	8.40	75	4.31	71

Tabelle 4: Betriebsdaten des Anaerobfilters. Dargestellt sind die organische Filterbelastung **L**, die Methankonzentration [CH₄] im gebildeten Gas, die spezifische Methanbildungsrate **P** und der prozentuale COD-Abbau COD_r in Abhängigkeit der **Verweilzeit**.

Verweilzeit Stunden	Probe	Acetat mM	Propionat mM	Butyrat mM	Isobutytrat mM
344	Zulauf	95.2	27.4	3.0	0.1
	Überlauf	0.6	0.2	0	0
110	Zulauf	85.1	24.0	1.9	0
	Überlauf	0.7	0.3	0	0
55	Zulauf	98.4	31.6	3.0	0.2
	Überlauf	1.2	0.3	0	0
44	Zulauf	103.0	33.7	4.5	0.3
	Überlauf	1.2	0.4	0	0
22	Zulauf	112.3	37.0	6.0	0.6
	Überlauf	1.4	0.5	0.1	0

Tabelle 5: Fettsäurekonzentrationen im Zulauf vom Substrat in den Anaerobfilter und im Anaerobfilterüberlauf.

hydraulische Retentionszeit
(Stunden)

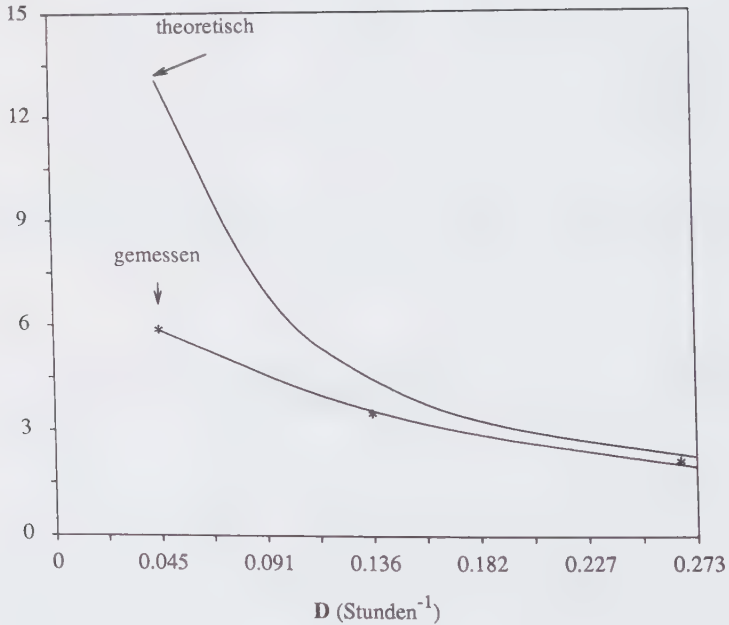


Abbildung 9: Tracerversuch.

Theoretische und gemessene Retentionszeiten für den Anaerobfilter. Dargestellt sind der Verlauf der theoretischen Retentionszeit und die gemessenen Retentionszeiten (*) in Abhängigkeit der Verdünnungsrate D .

4.1.3. Resultate mit der zweistufigen Anlage

Da sowohl der Fermenter wie auch der Anaerobfilter (Aufbau der Anlage siehe Abbildung 4) schon während Monaten im Einsatz waren, konnte ohne Anpassungsphase mit den Versuchen begonnen werden. Die Versuche mit dem vollständig durchmischten Fermenter hatten gezeigt, dass auf eine Rückführung der acetogenen Organismen in den Fermenter verzichtet werden konnte; die beiden Einheiten konnten daher direkt aneinander gekoppelt werden. Die organische Belastung im Anaerobfilter wurde während der 60-tägigen Versuchsdauer von $1.53 \text{ kg COD} \cdot \text{m}^{-3} \cdot \text{d}^{-1}$ auf $8.84 \text{ kg COD} \cdot \text{m}^{-3} \cdot \text{d}^{-1}$ (Abbildung 10) erhöht. Der pH-Wert sank dabei von 7.3, bei einer Retentionszeit von 110 Stunden, erst bei Senkung unter 44 Stunden Retentionszeit auf schliesslich 7.1 bei maximaler Filterbelastung. Bei gesteigerten Durchflussraten stieg die spezifische Methanbildungsrate innert drei bis fünf Tagen auf einen neuen Maximalwert an, und erreichte mit $4.21 \text{ m}^3 \text{CH}_4 \cdot \text{m}^{-3} \cdot \text{d}^{-1}$ den höchsten Wert. Die letzte Erhöhung der organischen Belastung von 5.36 auf $8.84 \text{ kg COD} \cdot \text{m}^{-3} \cdot \text{d}^{-1}$ bewirkte allerdings nur noch eine geringe Steigerung der spezifischen Methanbildungsrate von $0.04 \text{ m}^3 \text{CH}_4 \cdot \text{m}^{-3} \cdot \text{d}^{-1}$. Die Verringerung der Retentionszeiten führte zu einer Verminderung des COD-Abbaus von anfänglich 91% auf 69% (Tabelle 6), dabei sank die Methankonzentration im Gas von 77% auf 70%.

Verweilzeit Stunden	L $\text{kg COD} \cdot \text{m}^{-3} \cdot \text{d}^{-1}$	$[\text{CH}_4]$ %	P $\text{m}^3 \text{CH}_4 \cdot \text{m}^{-3} \cdot \text{d}^{-1}$	COD_r %
110	1.53	77	1.05	91
55	3.19	76	2.15	87
44	3.82	76	2.73	80
37	5.36	72	4.17	76
22	8.84	70	4.21	69

Tabelle 6: Betriebsdaten des Anaerobfilters bei vorgeschalteter Versäuerungsstufe. Dargestellt sind die organische Filterbelastung **L**, die Methankonzentration im gebildeten Gas $[\text{CH}_4]$, die spezifische Methanbildungsrate **P** und der prozentuale COD-Abbau COD_r in Abhängigkeit der Verweilzeit.

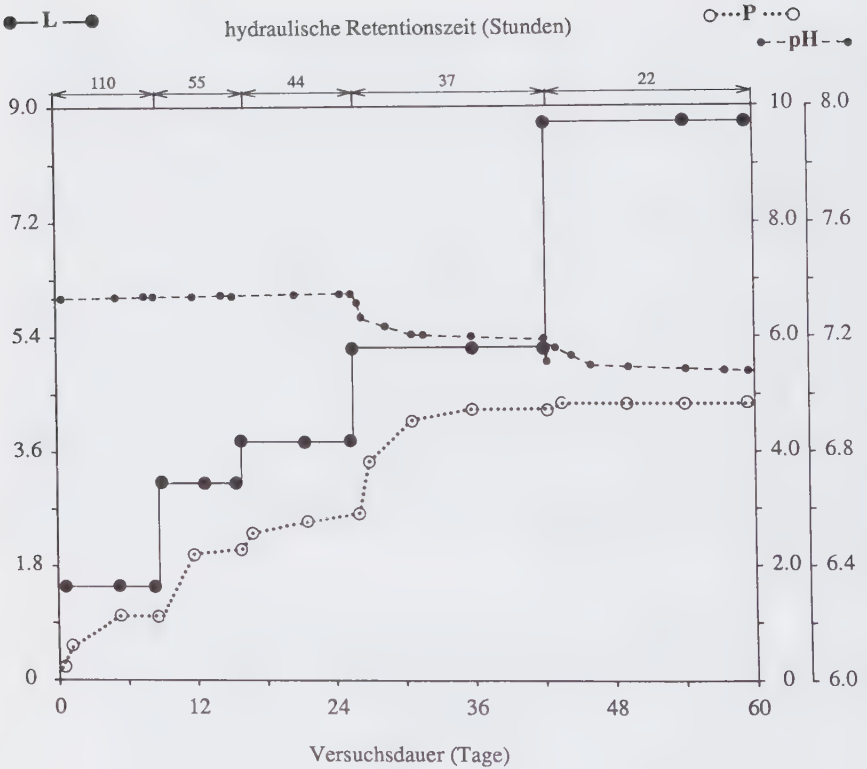


Abbildung 10: Zeitlicher Verlauf von Raumbelastung L (kg COD $\cdot$ m $^{-3}$ $\cdot$ d $^{-1}$), spezifischer Methanproduktionsrate P (m 3 CH $_4$ $\cdot$ m $^{-3}$ $\cdot$ d $^{-1}$) sowie des pH-Wertes als Funktion der Versuchszeit und der hydraulischen Retentionszeit als Parameter.

Im Filterüberlauf verschwand Propionat nie vollständig und Butyrat erschien wieder bei einer Retentionszeit von weniger als 44 Stunden (Tabelle 7). In der Versäuerungsstufe (Stufe 1) stieg die Belastung von 8.4 kg COD*m⁻³*d⁻¹ bei einer Retentionszeit von 20 Stunden auf 48.6 kg COD*m⁻³*d⁻¹ bei einer Retentionszeit von 4 Stunden. Tabelle 7 zeigt, dass in der Versäuerungsstufe vor allem Acetat und Propionat gebildet wurden. Die Butyrat- und Isobutyratkonzentrationen verdoppelten sich im Vergleich mit den Konzentrationen im Substrat, die Gesamtkonzentration blieb jedoch gering. Bei den Retentionszeiten von 7 und 4 Stunden war die Zunahme an Acetat und Propionat im Reaktor um ca. 40% geringer als bei den Retentionszeiten über 8 Stunden.

Die Fettsäurekonzentrationen im Substrat 2 zum Anaerobfilter änderten sich über die ganze Versuchsdauer nur gering und unabhängig von der Retentionszeit des Substrat 1 in der Versäuerungsstufe.

Verweilzeit Stunden	Probe	Acetat mM	Propionat mM	Butyrat mM	Isobutyrat mM
110	Zulauf	90.0	25.2	2.4	0.1
	Substrat 2	149.2	34.5	3.7	0.3
	Filter Überlauf	0.7	0.2	0	0
55	Zulauf	92.7	25.0	2.5	0.2
	Substrat 2	150.9	37.0	4.1	0.4
	Filter Überlauf	0.8	0.2	0	0
44	Zulauf	91.0	25.2	2.8	0.1
	Substrat 2	151.3	35.7	4.7	0.5
	Filter Überlauf	2.4	0.7	0.1	0
37	Zulauf	108.0	29.5	4.2	0.5
	Substrat 2	149.0	37.1	4.4	0.5
	Filter Überlauf	4.0	0.8	0.2	0
22	Zulauf	111.3	36.4	6.0	0.6
	Substrat 2	147.5	37.5	4.0	0.4
	Filter Überlauf	8.0	2.1	0.6	0

Tabelle 7: Fettsäurekonzentrationen der zweistufigen Anlage im Zulauf aus der Substratflasche in die Versäuerungsstufe, im Abfluss der Versäuerungsstufe in den Anaerobfilter (Substrat 2) und im Überlauf des Anaerobfilters. Die Verweilzeiten beziehen sich auf den Anaerobfilter.

4.1.4. Diskussion der Laborresultate

Der Einsatz eines vollständig durchmischten Fermenters lieferte nur bei Retentionszeiten von über 40 Stunden befriedigende Resultate, d.h., COD-Abbauleistungen von mindestens 70%. Anlagen dieser Art setzen ein sehr grosses Fermentervolumen voraus, um genügend grosse Abwassermengen effizient behandeln zu können. Die kürzeste Retentionszeit mit einer Abbauleistung über 70% lag bei 43 Stunden, was mit den kürzesten Retentionszeiten bei Grossanlagen übereinstimmt. Für den Anamet Process (Ab Sorigona, Schweden) werden minimale Retentionszeiten von 48 Stunden verlangt [26]. Bei Retentionszeiten unter 40 Stunden stiegen die Verluste an Biomasse im Fermenter stark an, vor allem hinsichtlich der Organismen der methanogenen Stufe, weil die vollständige Schlammrückgewinnung aus dem Überlauf unmöglich war. Dieser Biomasseverlust äusserte sich in einer reduzierten Gesamtabbauleistung und im Sinken der spezifischen Methanbildungsrate (Tabelle 2).

Die interessanteren Resultate im Rahmen der Laborversuche lieferte der Vergleich zwischen Anaerobfilter allein und Anaerobfilter mit vorgeschalteter homogen gemischter Versäuerungsstufe (zweistufige Anlage). Beim Anaerobfilter sank der pH-Wert bei jeder Substratzufuhrerhöhung (entspricht einem simulierten COD-Schock) über das ganze Filterprofil. Dies hatte jedoch keinen direkt messbaren Einfluss auf die spezifische Methanbildungsrate (Abbildung 8). *Bhatia* [27] erklärte dies aufgrund ähnlicher Versuche damit, dass die Methanproduktionsrate keine Aussage über den aktuellen Zustand der Organismen erlaube. Die Verdoppelungszeit der meisten acetoclastischen Methanogenen liegt über 10 Stunden. Deshalb können Messungen, welche in kleineren Zeiträumen nach der Schockbelastung vorgenommen werden, kaum zur Charakterisierung des physiologischen Zustandes des Systems herangezogen werden. Somit dürften die beobachteten kurzfristigen pH-Änderungen nur derartige Veränderungen in der Methanogenenpopulation bewirken, die sich erst später in einer reduzierten Methanproduktionsrate äusserten. Bei der zweistufigen Anlage traten solche pH-Sprünge erst bei Retentionszeiten unter 44 Stunden auf. Hier bildete die Versäuerungsstufe je nach Volumen einen Puffer für die auftretenden COD-Schocks. Die erhöhte organische Belastung erreichte den Filter nur noch in abgeschwächter Form und bewirkte eine Stabilisierung der spezifischen Methanbildungsrate nach einer Substratdurchflusserhöhung in ein bis zwei Tagen, wogegen sich der Anaerobfilter erst nach fünf Tagen stabilisierte (Abbildung 8, Abbildung 10).

Die Abbauleistung der Versäuerungsstufe von 10%-14% stimmt mit den Befunden anderer Autoren überein [28].

Die Versuche mit dem Anaerobfilter ergaben, dass nicht nur die Raumbelastung, sondern auch die Verdünnungsrate einen Einfluss auf die Abbauleistung hatte (Tabelle 4). Bei gleicher Raumbelastung tendierte die Betriebsweise mit der grösseren Verweilzeit zur besseren Abbauleistung, unabhängig vom Füllmaterial [29]. Die Fettsäurekonzentration im Anaerobfilterzulauf lag bei der zweistufigen Anlage um rund 50% höher, als bei der Einstufigen (Tabelle 5, Tabelle 7), ohne dass dabei die Propionsäure eine für die Methanogenese inhibitorische Konzentration erreichte [30, 31]. Vergleicht man jedoch die beiden Anlagen bezüglich spezifischer Methanproduktion und Abbauleistung, ergaben sich nur geringe Unterschiede (Tabelle 4, Tabelle 6).

Die Unterschiede zwischen theoretischer und gemessener Retentionszeit waren besonders bei den niedrigen Verdünnungsraten erheblich (Abbildung 9). Offensichtlich durchströmt das Substrat dabei einige bevorzugte Kanäle im Filter und verteilt sich nicht homogen über den ganzen Filterquerschnitt, was schon früher auch von *Genung* [32] beobachtet worden war. Homogenere Strömungseigenschaften erreichte *Aivasidis* [33], indem er in mehreren Versuchen verschiedene Materialien bezüglich ihrer Eignung als Matrix testete. Der Einsatz eines geeigneten Trägermaterials ermöglichte ihm, den Filter mit einer maximalen Raumbelastung von $102 \text{ kg COD} \cdot \text{m}^{-3} \cdot \text{d}^{-1}$ zu betreiben.

Da der COD-Abbau direkt mit der Methanbildung korreliert ist und Verluste durch Bildung von H_2S oder H_2 vernachlässigbar sind [34], wurde für die stationären Zustände eine einfache Methanbildungskinetik nach *Chen* [35] berechnet (Herleitung im Anhang). Tabelle 8 zeigt, dass für den vollständig durchmischten Fermenter der theoretische Ausbeutekoeffizient Y_0 gut mit dem gemessenen übereinstimmt. Auch die theoretische minimale Retentionszeit HRT_{th} stimmt mit dem Versuchsergebnis überein (Abbildung 7). Für die beiden Anaerobfiltersysteme dagegen liegen die gemessenen Ausbeutekoeffizienten deutlich unter den theoretischen Werten. Die minimale Retentionszeit konnte experimentell nicht ermittelt werden, da vorher eine Übersäuerung des Filters stattfand. Bei Versuchen mit pH-Regelung und verdünntem Substrat konnte die Retentionszeit auf 10 Stunden verkürzt werden, ohne die kritische Verdünnungsrate bezüglich Auswaschung der Methanogenen erreicht zu haben. Dies unterstützte die Vermutung, dass bei höheren Verdünnungsraten die Durchströmung im Filter homogener war (Abbildung 9).

	Y_o	Y_{gem}	μ_{max}	HRT_{th}	HRT_{gem}
<i>Fermenter</i>	0.51	0.53	0.114	42	43
<i>Anaerobfilter</i>	0.40	0.57	0.299	9	nb
<i>zweistufige Anlage</i>	0.42	0.58	0.304	8	nb

Tabelle 8: Vergleich der theoretischen und die durch Versuche bestimmten kinetischen Konstanten für die Methanbildung. Die Ausbeutekoeffizienten beziehen sich auf Gasvolumen im Normzustand: 273,15 °K ; 101 kPa.

Abkürzungen:

- Y_o : Maximaler theoretischer Ausbeutekoeffizient ($m^3CH_4/kg\ COD_{in}$).
- Y_{gem} : Experimentell ermittelter maximaler Ausbeutekoeffizient ($m^3CH_4/kg\ COD_{in}$).
- HRT_{th} : Minimale theoretische Retentionszeit (Stunden).
- HRT_{gem} : Experimentell ermittelte minimale Retentionszeit (Stunden).
- μ_{max} : Maximale spezifische Wachstumsrate (Tag^{-1}).
- nb: nicht bestimmt.

4.2. Betrieb der Pilotanlage in der Zuckerfabrik Frauenfeld

Die Versuche verteilten sich auf die drei Zuckerrüben-Verarbeitungskampagnen der Jahre 1981 bis 1983, jeweils von Oktober bis Dezember. Da in den ersten zwei Jahren die Versuche oft durch technische Pannen oder durch Umgestaltungen der Versuchsanlage unterbrochen wurden, liegt der Schwerpunkt der Auswertungen in den Resultaten des Jahres 1983. In jenem Jahr war die Pilotanlage erstmals ohne Pannen und Unterbrüche durchgehend in Betrieb. Die Darstellung der Ergebnisse der Jahre 1981 und 1982 haben mehr protokollarischen Charakter und sollen die Probleme beim Betrieb des Filters in grösserem Massstab dokumentieren.

4.2.1. Kampagne 1981

Am 1.10. wurde der Fermenter angeimpft, am 5.10. begann man mit dem Durchpumpen von Schwemmwasser mit einer Verdünnungsrate von 0.02/Stunde. Bis zum 30.10. traten keine Pannen im Betriebsablauf auf. Da bereits fünf Tage nach dem Animpfen mit dem Durchpumpen von Schwemmwasser begonnen worden war (beim Laborfilter betrug diese Wartezeit zwei Wochen), erreichte die Pilotanlage bei einer Retentionszeit von 35 Stunden nach 17 Tagen Betriebsdauer eine Abbauleistung von lediglich 58% und einen Ausbeutekoeffizienten von $0.31 \text{ m}^3 \text{CH}_4/\text{kg COD}$. Von da an erschwerte vor allem die mangelhafte mechanische Belastbarkeit des für den Pumpenschlauch verwendeten Materials eine über längere Zeit kontinuierliche Prozessführung. Im Pumpenkopf traten am Schlauch wiederholt Risse auf. Am 26.11. ersetzte man den Silikonschlauch mit einem solchen aus Viton, der sich aber als noch weniger robust erwies. Der nach dem 1.12. eingesetzte Neoprenschlauch bewährte sich bis zum Versuchsende. In dieser zweiten pannenfreien Phase verbesserte sich die Abbauleistung bei einer Retentionszeit von 30 Stunden auf 66%, der Ausbeutekoeffizient stieg auf $0.41 \text{ m}^3 \text{CH}_4/\text{kg COD}$. Messungen ergaben, dass die COD-Reduktion im vorgeschalteten Ausgleichsgefäss bis 11% erreichte. Deshalb entnahm man die Proben für den Zufluss nach dem Ausgleichsgefäss und übernahm nicht die Laborwerte vom Schwemmwasserkreislauf für die Berechnungen. Kurz vor dem Ende der Kampagne spülte man den Filter zur Aufbewahrung bis zur nächsten Saison von oben mit Schwemmwasser durch und verschloss ihn dann, vollständig aufgefüllt, möglichst luftdicht ab.

Tabelle 9 gibt eine Übersicht der Betriebsdaten von 1981.

Die Erfahrungen mit der Pilotanlage liessen für die kommenden Kampagnen folgende Schlüsse zu:

- Nach dem Animpfen müssen mehr als fünf Tage gewartet werden, bevor mit dem Durchpumpen von Schwemmwasser begonnen werden kann.
- Da die Abbauleistung des Filters beim Herabsetzen der Retentionszeit von 35 auf 30 Stunden zunahm, darf die Retentionszeit noch weiter herabgesetzt werden.

V Stunden	L kg COD*m ⁻³ *d ⁻¹	[CH ₄] %	P m ³ CH ₄ *m ⁻³ *d ⁻¹	COD _r %	Y m ³ CH ₄ /kg COD
35	8.0	64	2.94	58	0.31
30	14.0	67	6.92	66	0.41

Tabelle 9: Betriebsdaten der Pilotanlage in der ZFF im Jahre 1981. Dargestellt sind die organische Filterbelastung **L**, die Methankonzentration im gebildeten Gas [CH₄], die spezifische Methanbildungsrate **P**, der prozentuale COD-Abbau COD_r, sowie der Methanausbeutekoeffizient **Y** als Funktion der Verweilzeit **V**.

4.2.2. Kampagne 1982

Das Animpfen des Filters fiel weg, da er am Ende der letzten Kampagne nicht geleert worden war. Am 1.10. begann man mit dem Durchpumpen von Schwemmwasser mit einer Verdünnungsrate von 0.02/Stunde. Am 3.10 betrug die Gasproduktion schon 50 l/Stunde. Am 8.10. stellte man fest, dass der Pumpenkopf den Schlauch zu wenig zusammenpresste und die tatsächliche Verdünnungsrate lediglich 0.012 statt 0.02/Stunde betrug. Nach der Korrektur stieg die spezifische Methanbildungsrate auf 3.76 m³CH₄*m⁻³*d⁻¹, der COD-Abbau betrug dabei 73%. Danach verkürzte man die Retentionszeit auf 25 Stunden. Die spezifische Methanbildungsrate erhöhte sich in 15 Tagen auf 8.13 m³CH₄*m⁻³*d⁻¹, der COD-Abbau verringerte sich dabei nur gering auf 71%. Der Ausbeutekoeffizient fiel von 0.47 auf 0.40 m³CH₄/kg COD. Nach dem Erhöhen der Verdünnungsrate am 26.10 auf 0.07/Stunde riss erneut der Pumpschlauch. Am 8.11. verstopfte eine Leccakugel den Überlauf und füllte den Gaszähler mit Schwemmwasser. Ab dem 19.11. stieg der Druck am Pumpenschlauch von den normalen 0.5 atm kontinuierlich, bis er am 3.12. mit 1.25 atm den höchsten Wert erreichte. Die

Gasproduktion nahm mit steigendem Druck ab. Sie stieg jedoch sprunghaft wieder an, wenn die Substratzufuhr gestoppt wurde und der Druck sich erneut auf 0.5 atm normalisiert hatte. Das anschliessend aufgenommene Fettsäureprofil deutete auf eine Filterverstopfung zwischen den Probeentnahmestellen E3 und E5 hin (Abbildung 11). In der Zeit vor dem Druckanstieg stieg die Filterbelastung auf einen Maximalwert von 33 kg COD*m⁻³*d⁻¹. Die Abbauleistung verringerte sich auf 68%, der Ausbeutekoeffizient auf 0.26 m³CH₄/kg COD. Alle Versuche, den Filter durch Waschen mit Schwemmwasser von oben oder unten zu entstopfen, bewirkten keine Veränderung der Druckverhältnisse oder der Fettsäureprofile. Am 11.12. beendete man die Versuche und wusch den Filter gründlich mit Wasser durch, um später eine Untersuchung der Füllung zu ermöglichen. Tabelle 10 fasst die Betriebsdaten bis zum 15.11., also kurz vor dem Druckanstieg zusammen. Nach der Kampagne ergab eine Untersuchung der Filterfüllung, dass im Bereich der Probeentnahmestellen E2, E3, E4 und E5 sich ein kompakter Zapfen aus Leccakugelbruchstücken, Lehm vom Schwemmwasser und aus Klärschlamm gebildet hatte. Die restliche Füllung bestand aus Aggregaten von 5-10 cm Durchmesser. Sogar entlang der Reaktorwand hatte sich eine bis 1cm dicke Lehmschicht abgelagert.

Die Folgerungen für die nächste Kampagne waren:

- Das Füllen des Reaktors mit dem Trägermaterial muss möglichst schonend geschehen.
- Die Säule muss nach dem Füllen soweit wie möglich durch Waschen von Bruchstücken gereinigt werden.
- Der Filter darf nur mit kleinen Mengen Klärschlamm angeimpft werden.

V Stunden	L kg COD*m ⁻³ *d ⁻¹	[CH ₄] %	P m ³ CH ₄ *m ⁻³ *d ⁻¹	COD _r %	Y m ³ CH ₄ /kg COD
50	6.6	69	3.76	73	0.47
25	17.0	68	8.13	71	0.40
17	33.0	69	10.29	68	0.26

Tabelle 10: Betriebsdaten der Pilotanlage in der ZFF im Jahre 1982. Dargestellt sind die organische Filterbelastung **L**, die Methankonzentration im gebildeten Gas **[CH₄]**, die spezifische Methanbildungsrate **P**, der prozentuale COD-Abbau **COD_r**, sowie der Methanausbeutekoeffizient **Y** als Funktion der Verweilzeit **V**.

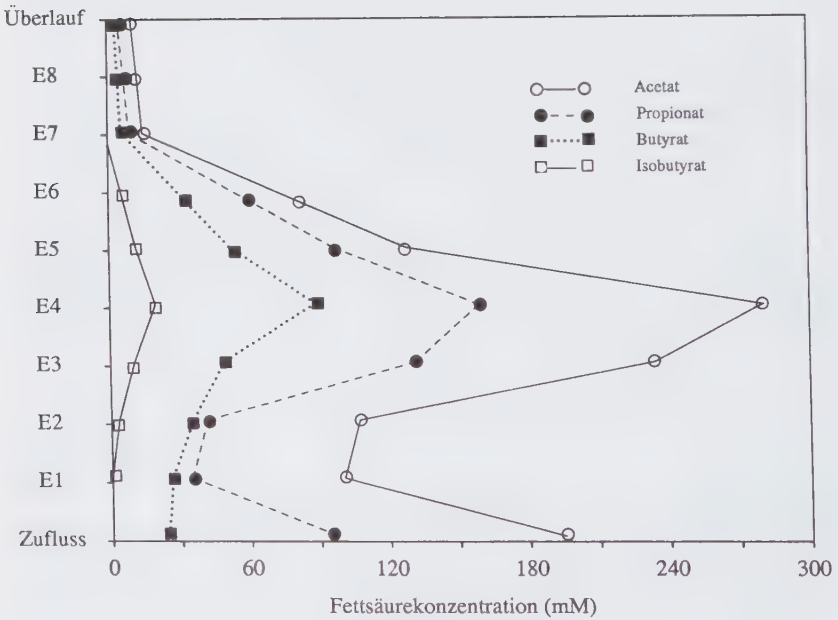


Abbildung 11: Verteilung der Fettsäurekonzentrationen (C_2 - C_4) über das Höhenprofil der Pilotanlage bei vermuteter Filterverstopfung im Bereich zwischen den Probeentnahmestellen E3 und E4 während der Kampagne 1982.

4.2.3. Kampagne 1983

Aufgrund der 1982 aufgetretenen Verstopfungsprobleme erfolgte das Auffüllen und Animpfen des Filters nach der in Material und Methoden beschriebenen Art. Nach dem Animpfen des Filters erreichte die Gasbildungsrate nach acht Tagen ohne Schwemmwasserdurchfluss mit 54 l/Stunde das Maximum. Darauf begann man mit dem Durchpumpen von Schwemmwasser. Abbildung 12 zeigt die drei Versuchsphasen während der 80-tägigen Betriebsdauer. Der COD-Gehalt des Schwemmwassers stieg in 40 Tagen bis zu einem Maximalwert von 28 g/l an, nachher reduzierten Regenfälle und zudosiertes Frischwasser den Wert auf 16 - 22 g/l.

Die erste Phase diente dazu, die methanbildenden Organismen zu einer möglichst dichten Population anwachsen zu lassen. In dieser Zeit stieg die Filterbelastung auf $8.1 \text{ kg COD} \cdot \text{m}^{-3} \cdot \text{d}^{-1}$ mit einer Abbauleistung von 76% und einem Ausbeutekoeffizienten von 0.62 (Tabelle 11). Wie Abbildung 16 zeigt, fand dabei die grösste COD-Reduktion im untersten Filterteil statt. Oberhalb der Probeentnahmestelle E2 blieb der COD-Wert konstant.

Da die vorhergehenden Kampagnen gezeigt hatten, dass die maximale Filterbelastung zwischen 20 und 30 $\text{kg COD} \cdot \text{m}^{-3} \cdot \text{d}^{-1}$ und die optimale Retentionszeit zwischen 25 und 17 Stunden lagen, verkürzte man die Retentionszeit in der zweiten Phase auf 23 Stunden. Sogar bei der maximalen Filterbelastung von 20 $\text{kg COD} \cdot \text{m}^{-3} \cdot \text{d}^{-1}$ betrug die COD-Abbauleistung 73%. Dabei erreichte die spezifische Methanproduktionsrate einen Wert von $10.86 \text{ m}^3 \text{CH}_4 \cdot \text{m}^{-3} \cdot \text{d}^{-1}$ mit einem Ausbeutekoeffizienten von 0.45 (Tabelle 11). Aus dem COD-Profil bei der Retentionszeit von 23 Stunden ist in Abbildung 17 ersichtlich, dass trotz der dabei auftretenden hohen Belastungen die COD-Reduktion hauptsächlich immer noch im unteren Filterteil stattfand, der COD-Gehalt jedoch erst oberhalb der Probeentnahmestelle E5 konstant blieb. Gleichzeitig ereignete sich während dieser Periode mit den höchsten COD-Werten im Zufluss ein Temperatursturz von 6°C (Abbildung 13), ohne dass dies einen messbaren Einfluss auf die Reinigungseffizienz des Filters gehabt hätte.

Abbildung 12 zeigt, dass in der dritten Phase bei einer Retentionszeit von 17 Stunden sowohl die spezifische Methanbildungsrate wie auch die Abbauleistung der Pilotanlage abnahm, dies trotz sinkender COD-Werte im zugeführten Schwemmwasser (Tabelle 11). Aus Abbildung 17 ist ersichtlich, dass sich der Abbau der organischen Substanzen über die ganze Filterhöhe verteilte und der COD-Gehalt des Abwassers erst vor dem Überlauf einen konstanten Wert annahm. Vergleich man die Fettsäureprofile (Abbildung 16) mit jenen bei Retentionszeiten von 50 bzw. 23 Stunden (Abbildungen 14 und 15), stellte man wiederum einen Abbau der mit dem Schwemmwasser zugeführten Fettsäuren im untersten Reaktorteil fest. Die Neubildung jedoch verschob sich mit abnehmender Retentionszeit im Filter nach oben. Dies änderte sich aber auch dann nicht weiter, als gegen Ende der Kampagne die COD-Werte erneut anstiegen und die Filterbelastung mit $29.4 \text{ kg COD} \cdot \text{m}^{-3} \cdot \text{d}^{-1}$ ein absolutes Maximum erreichte.

V Stunden	L $\text{kg COD} \cdot \text{m}^{-3} \cdot \text{d}^{-1}$	$[\text{CH}_4]$ %	P $\text{m}^3 \text{CH}_4 \cdot \text{m}^{-3} \cdot \text{d}^{-1}$	COD_r %	Y $\text{m}^3 \text{CH}_4 / \text{kg COD}$
50	8.1	70	6.08	76	0.62
23	20.0	68	11.07	73	0.45
17	29.4	64	10.86	69	0.30

Tabelle 11: Betriebsdaten der Pilotanlage in der ZFF im Jahre 1983. Dargestellt sind die organische Filterbelastung **L**, die Methankonzentration im gebildeten Gas $[\text{CH}_4]$, die spezifische Methanbildungsrate **P**, der prozentuale COD-Abbau COD_r sowie der Methanausbeutekoeffizient **Y** als Funktion der Verweilzeit **V**.

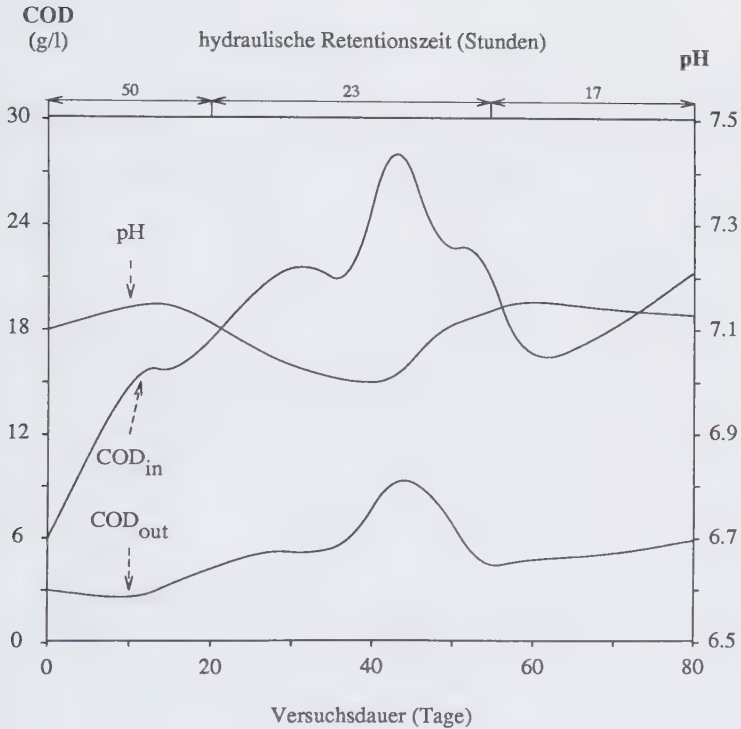


Abbildung 12: Dargestellt sind die starken Schwankungen der COD-Konzentration im zugeführten Schwemmwasser, die zeitlichen Veränderungen der COD-Konzentration des Schwemmwassers am Filterüberlauf und der pH-Wert des behandelten Abwassers am Filterüberlauf bei Variation der hydraulischen Retentionszeit während der Kampagne 1983.

Temperatur
(°C)

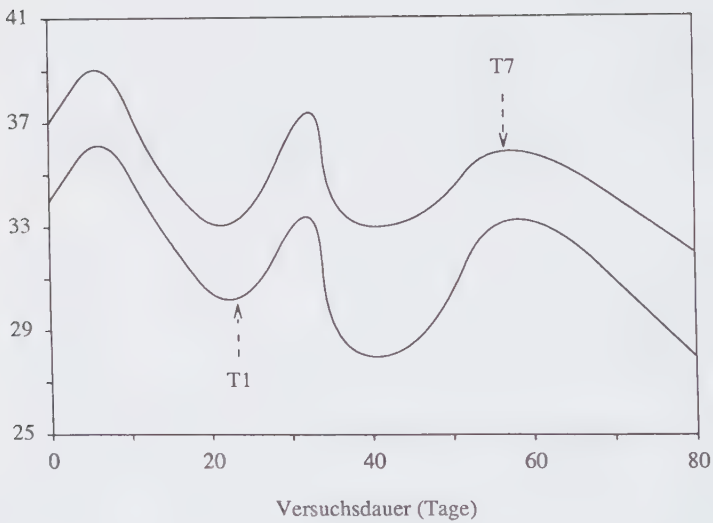


Abbildung 13: Zeitlicher Temperaturverlauf. Dargestellt ist das Temperaturgefälle im Filter zwischen der untersten (T1) und obersten Pt100 Temperatursonde (T7) im Filter während der Kampagne 1983.

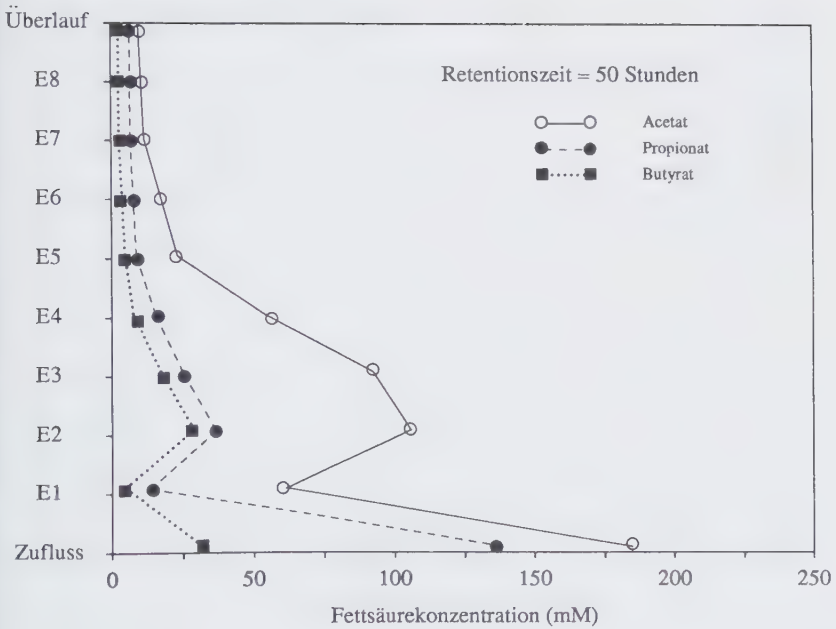


Abbildung 14: Verteilung der Fettsäurekonzentrationen (C_2 - C_4) im Filterwasser über das Höhenprofil des Filters bei einer hydraulischen Retentionszeit von 50 Stunden. Die Bestimmung geschah vor dem Herabsetzen der hydraulischen Retentionszeit auf 23 Stunden.

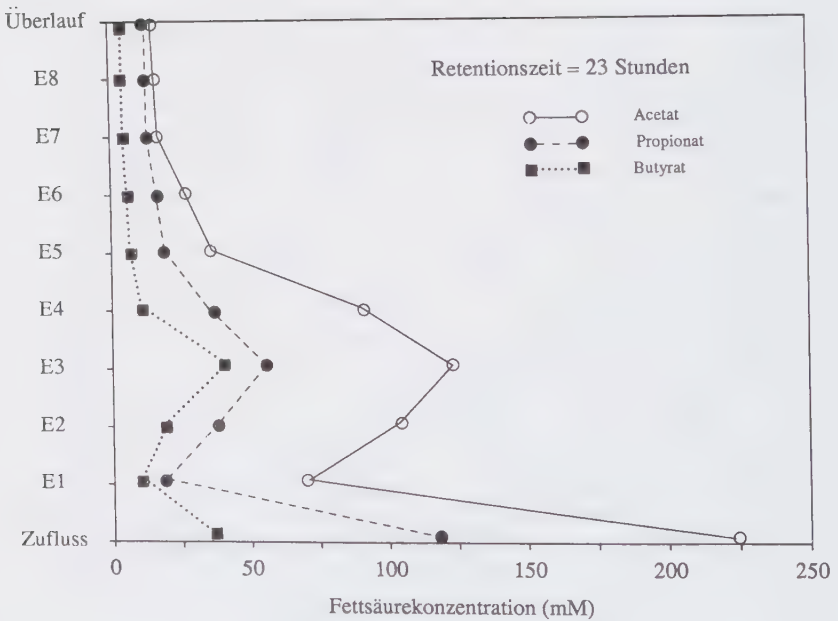


Abbildung 15: Verteilung der Fettsäurekonzentrationen (C_2 - C_4) im Filterwasser über das Höhenprofil des Filters bei einer hydraulischen Retentionszeit von 23 Stunden. Die Bestimmung geschah vor dem Herabsetzen der hydraulischen Retentionszeit auf 17 Stunden.

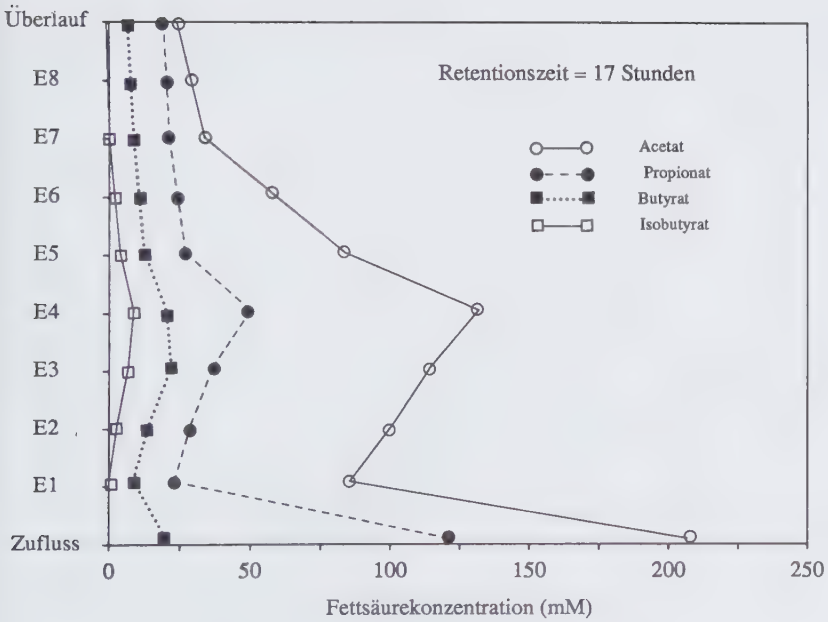


Abbildung 16: Verteilung der Fettsäurekonzentrationen ($C_2 - C_4$) im Filterwasser über das Höhenprofil des Filters bei einer hydraulischen Retentionszeit von 17 Stunden.

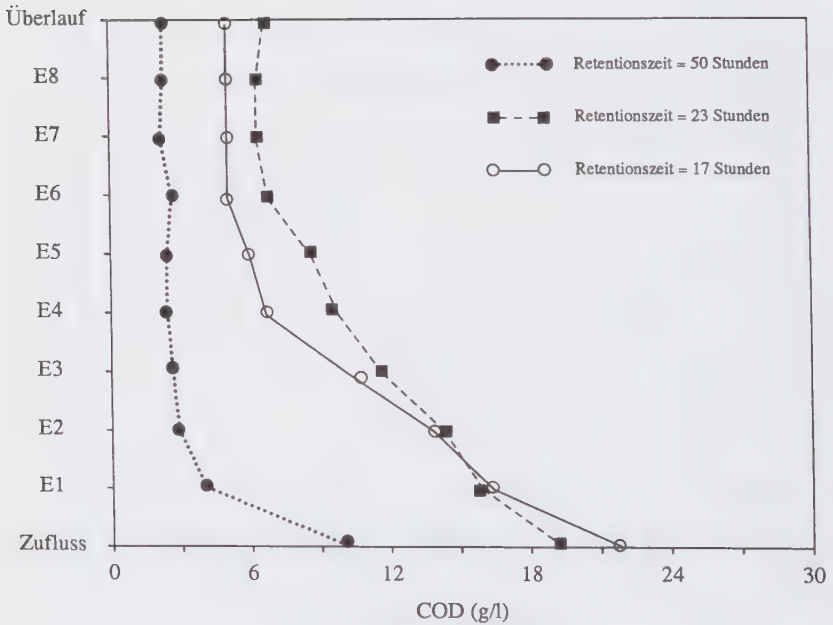


Abbildung 17: COD-Profile im Filterwasser bei den drei Retentionszeiten von 50, 23 und 17 Stunden, bzw. organischen Filterbelastungen von 8.1, 20.0 und 29.4 kg COD*m⁻³*d⁻¹. Die Profile wurden vor der jeweiligen Erhöhung der Filterbelastung untersucht.

4.2.4. Diskussion der Resultate mit der Pilotanlage

Das Scaling-up des Anaerobfilters zur Pilotanlage war mit einem Volumenfaktor 60 möglich. Diese Volumenvergrößerung zeigte jedoch vor allem in den ersten beiden Kampagnen, dass die Methodik den neuen Filterdimensionen angepasst werden musste. Dies betraf in erster Linie das Füllen und das Animpfen des Filters. Generell war die Beanspruchung aller eingesetzten Geräte um ein Vielfaches grösser als im Labor. Als problematisch erwies sich die Bestimmung der Gasaustrittsrates mit dem Gaszähler. Bei Gasflüssen bis zu 400 l/Stunde konnte das Wasservolumen im Zähler nie konstant gehalten werden, da das eintretende Gas mit Wasser gesättigt war und dem Gaszähler ständig neues Wasser zuführte. Der daraus resultierende Messfehler pflanzte sich in allen weiterführenden Leistungs- und Energieberechnungen fort.

Faulschlamm als Impfmateriel benötigte nur eine kurze Adaptationsphase von ca. 10 Tagen an das Rübenschwemmwasser; eine kurze Zeit verglichen mit den Adaptationszeiten, die bei anderen Substraten [36] gefunden wurden.

Das Ausgleichsgefäß erwies sich als primitive Versäuerungsstufe. Die Fettsäurekonzentrationen im Schwemmwasser zwischen Ausgleichsgefäß und Filter waren bis zu 65% höher als im Schwemmwasserkreislauf. Da die Versäuerung jedoch unkontrolliert und nicht in Richtung solcher Metabolite verlief, die von den acetoclastischen Methanbakterien direkt verwendet werden konnten, gingen im Ausgleichsgefäß Reduktionsäquivalente in Form von H_2 verloren [37]. Die gemessene COD-Abnahme im Ausgleichsgefäß von bis zu 10% bei einem pH von 5.5-6.0 unterstützt diese Annahme.

Die optimale organische Belastung für eine maximale spezifische Methanproduktionsrate lag zwischen 20 und 29 kg COD*m⁻³*d⁻¹, die dabei erreichte Methanausbeute schwankte zwischen 0.3 und 0.45 m³CH₄/kg COD. Verglichen mit der Laboranlage konnte die Pilotanlage somit bei einer beinahe verdreifachten organischen Belastung betrieben werden. Vermutlich war die Volumenausnutzung bei der Pilotanlage, bedingt durch die grösseren Leccakugeln, den veränderten Druckverhältnissen im Filter und einer erhöhten Gasbildungsrate, weit besser als bei der Laboranlage. Die Abbildungen 14 bis 17 zeigen, dass bei steigenden organischen Belastungen das ganze Filtervolumen ausgenutzt wurde, indem sich die Methanogenen immer weiter nach oben verlagerten. Bei einer organischen Belastung von bis zu 30 kg COD*m⁻³*d⁻¹ betrug die kürzeste Retentionszeit ca. 17 Stunden. Eine weitere Erhöhung der Verdünnungsrate hätte dem

COD-Profil in Abbildung 17 nach zu urteilen den Filter übersäuert.

Leccakugeln als Trägermaterial erwiesen sich als ungeeignet hinsichtlich eines weiteren Scale-up, da das Verhältnis der durch Organismen nutzbaren Oberfläche zum Kugelvolumen geringer als bei anderen eingesetzten Materialien war [38, 39, 40]. Dass mit geeigneten Trägermaterialien die Filterbelastung bis auf $100 \text{ kg COD} \cdot \text{m}^{-3} \cdot \text{d}^{-1}$ gesteigert werden kann, bewies schon Aivasidis [33].

4.2.5. Energetische Aspekte für eine Grossanlage

Für die Reinigung eines Teiles der in einer Zuckerfabrik täglich anfallenden Menge an Rübenschwemmwasser von bis zu 1000 m^3 [41] würde ein Filter von $100\text{-}150 \text{ m}^3$ benötigt. Das Ziel beim Betrieb einer solchen Anlage muss aus Rentabilitätsgründen nicht nur die hohe Reinigungsleistung, sondern auch ein möglichst hoher Nettoenergiegewinn in Form von Methan sein.

Die Nettoenergiemenge ergibt sich dabei aus der Differenz der von der Anlage in Form von Methan erzeugte Bruttoenergie und der für den Prozesseigenbedarf benötigten Energiemenge. Den überwiegenden Anteil am Prozesseigenbedarf bildet die Energie zur Erwärmung des Zulaufs auf die Betriebstemperatur und die Thermostatisierung des Filters [42]. Konstruktive Massnahmen, die den Prozessenergiebedarf verringern, sind demnach die direkte Einspeisung des Schwemmwassers in den Filter ohne Zwischenspeicherung und die Verminderung von Transmissionsverlusten durch Isolation. Da die Zuckerrübenverarbeitung in den Wintermonaten stattfindet, beträgt der Unterschied der Prozesstemperatur zur Umgebungstemperatur bis zu 40°C . Eine Anlage von 100 m^3 könnte mit den Leistungsdaten der Pilotanlage eine maximale Bruttoenergie/Tag von 11000 kWh/Tag erzeugen. Erfahrungsgemäss liegt der durchschnittliche Prozesseigenbedarf an Energie/Tag für eine solche Anlage zwischen 5000 und 6500 kWh/Tag . Die verbleibenden 4500 bis 6000 kWh/Tag stünden als Nettoenergie/Tag zur Verfügung, was je nach Wirkungsgrad der Umwandlung (Gasheizung oder Gasmotor-Elektrogenerator), etwa $2\text{-}3\%$ des täglichen Energiebedarfs einer Zuckerfabrik entspricht.

4.3. Prozessoptimierung mit Hilfe eines Rechners

Der Aufbau der Versuchsanordnung ist in Abbildung 5 (Material und Methoden) dargestellt.

4.3.1. Die Struktur des Optimierungsprogrammes.

An on-line Information über den Anaerobfilter stehen für eine Optimierungsstrategie pH-Wert, Gasaustrittsrate und Methankonzentration im gebildeten Gas zur Verfügung. Mit dem PCS68000 stand bereits ein leistungsfähiges Prozessdatenerfassungs- und Kontrollprogramm zur Verfügung, welches mit Hilfe von Interfacefunktionen auf einfache Weise Informationen mit anderen Programmen austauschen kann.

Das OS9-Betriebssystem ermöglichte eine Zerlegung der Optimierungsstrategie in mehrere kleine Einzelprogramme, wodurch unübersichtliche und in der Fehlersuche zeitraubende Programmstrukturen vermieden werden konnten (Abbildung 18). Das eigentliche Optimierungsprogramm simulierte einen Operator, der dank seiner Kenntnisse der Betriebsparameter Entscheidungen zur Stabilisierung der Prozessführung traf. Die Software bestand aus sechs im Hintergrund laufenden und zwei mit dem Benutzer interaktiv kommunizierenden Programmen.

Hintergrundprogramme:

"Datatransfer"

Dieses Programm initialisierte die Regelparameter mit Defaultwerten und das Druckerprogramm *"Prozessmeldungen"*. Anschliessend führte es den kontinuierlichen Datentransfer zwischen dem PCS68000-Datenfeld und dem Prozessoptimierungs-Datenfeld unter Berücksichtigung eventueller pH-Eichvorgänge durch.

"Mittelwerte"

Die Bestimmung der Mittelwerte von pH- und Gasaustrittsrate fand in frei wählbaren Zeitintervallen statt. Das Programm speicherte auch die Änderung der letzten 20 pH-Mittelwerte ab und berechnete die aktuelle Methanflussrate aus der Gasaustrittsrate und der Methankonzentration im Gas.

"Gasanalyse"

Da der Horiba Gasanalysator wechselweise den Methangehalt von zwei verschiedenen Gasquellen bestimmen musste, kontrollierte dieses Programm die fünf Ventile der Gasumschaltanlage.

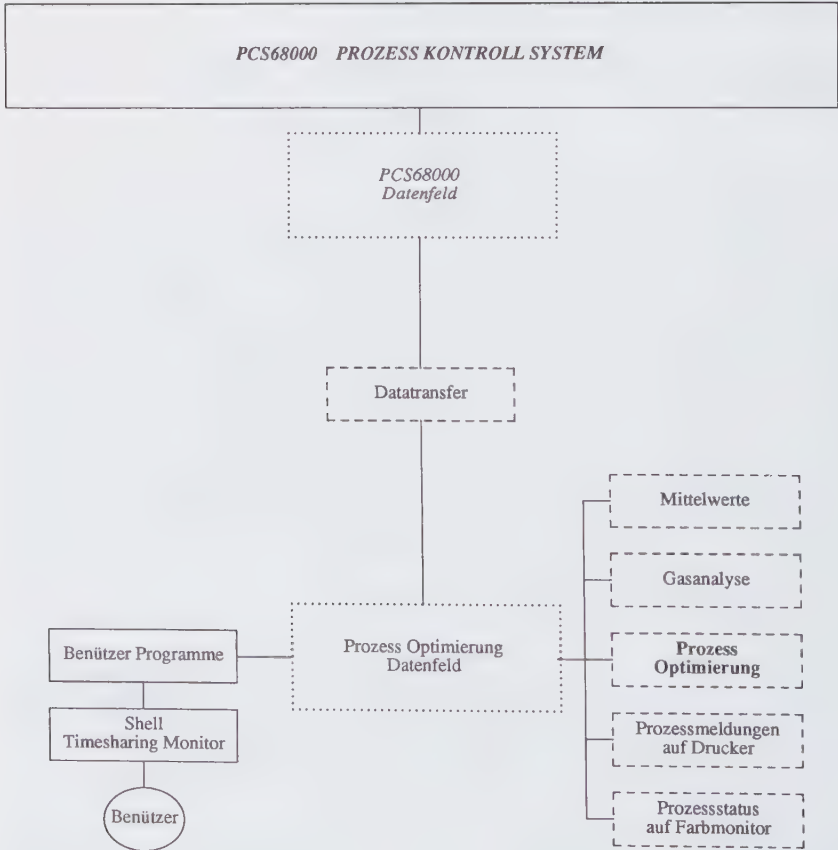


Abbildung 18: Struktur der Optimierungsprogramme. Die ausgezogenen Kästchen sind interaktiv arbeitende Programme, die gestrichelten Kästchen stellen im Hintergrund laufende Programme dar, gepunktet sind die gemeinsamen Datenfelder symbolisiert.

Nach jedem Umschalten verstrich eine gewisse Zeit, bis sich die Messküvette vollständig mit dem neuen Gas gefüllt hatte. Diese Wartezeit konnte on-line angepasst werden, da sie vom aktuellen Gasfluss der jeweiligen Gasquelle abhing. Anschliessend berechnete das Programm den Mittelwert der Methankonzentration über ein gewünschtes Zeitintervall hinweg.

"Prozessmeldungen"

Alle Meldungen der Hintergrundprogramme über Prozessänderungen und eventuelle Störungen erfolgten über ein Druckerprogramm, das die von den Hintergrundprogrammen ausgesandten Signale decodierte und eine entsprechende Meldung an den Drucker weiterleitete. Mit diesem Vorgehen sollte verhindert werden, dass wichtige Kontrollprogramme wegen eines Druckerdefektes blockiert blieben.

"Prozessstatus"

Dieses Graphikprogramm stellte alle aktuellen Mess- und Regelgrössen in einem Anlageschema auf dem Farbmonitor dar.

"Prozessoptimierung"

Dieses eigentliche Optimierungsprogramm regelte die Substratzufussrate in den Anaerobfilter durch die Wahl einer entsprechenden Pumpleistung. Das Flussdiagramm dieses Programms ist in Abbildung 19 dargestellt. Um kritische pH-Änderungen auch ausserhalb der Entscheidungsintervalle ($\delta_{T_{int}}$) erfassen zu können, fand ständig ein Vergleich zwischen dem aktuellen (pH_{akt}) und einem vom Benutzer als kritisch definierten pH-Wert ($\delta_{pH_{krit}}$) statt. Sank der aktuelle unter den kritischen pH-Wert, stoppte die Substratzufuhr, bis der pH erneut zu einem definierten Wert ($\delta_{pH_{rest}}$) angestiegen war. Damit dieser Unterbruch der Substratzufuhr nur in äussersten Notfällen auftrat, versuchte man durch Beobachtung der zeitlichen pH-Änderungen im voraus auf Ereignisse im Filter zu reagieren. Überschritt die pH-Abnahmerate einen kritischen Wert ($\delta_{\Delta pH_{krit}}$), korrigierte das Programm die Zufussrate um einen Betrag (δ_{P_2}) und wartete eine Zeitspanne (T_{sleep}) bis zur nächsten Korrektur. In den Entscheidungsintervallen selbst bestimmte nicht nur die pH-Änderung (ΔpH) die Korrektur der Zufussrate. Die Wahl des Korrekturfaktors ($\delta_{P_1} - \delta_{P_6}$) hing auch von der Änderung der Methankonzentration im gebildeten Gas (ΔCH_4_{konz}) und des Methanflusses (ΔCH_4_{fluss}) ab. In Abbildung 19 ist dies der Übersicht wegen nicht dargestellt.

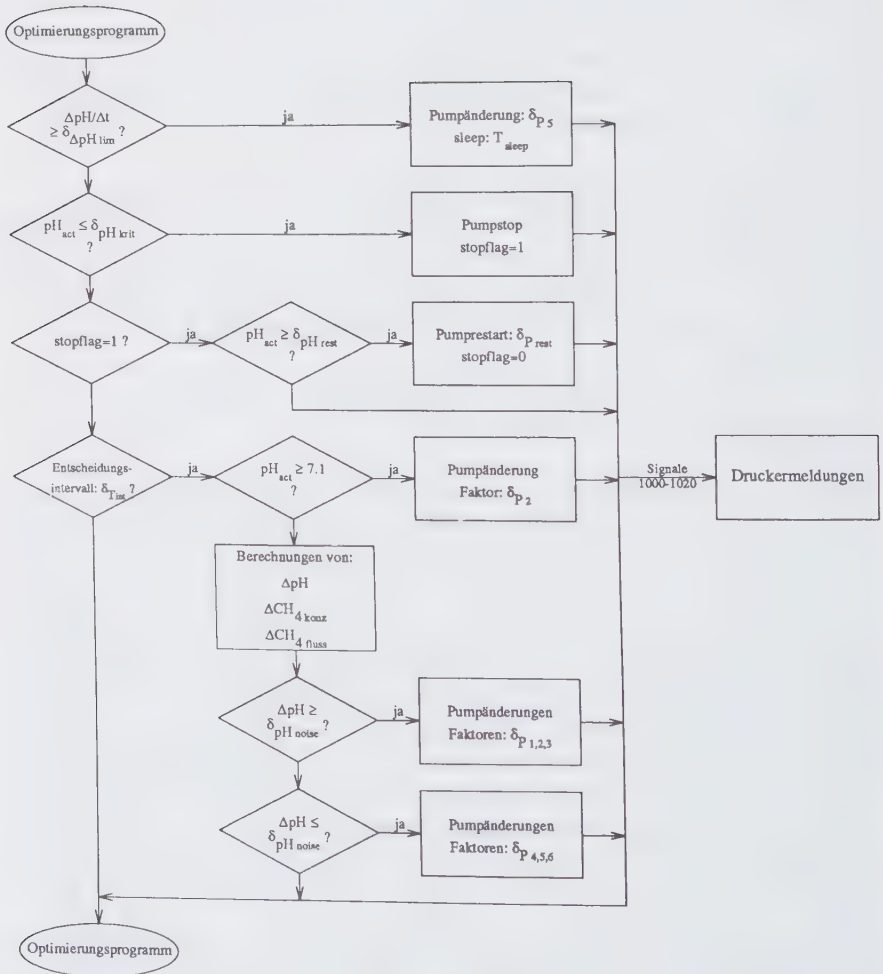


Abbildung 19: Flussdiagramm des Optimierungsprogrammes. Alle Regelgrößen, welche vom Benutzer on-line verändert werden können, sind im Flussdiagramm mit δ bezeichnet.

Interaktive Programme:

"Zeitvariablen"

Mit diesem Programm konnte der Benutzer für alle zeitabhängigen Parameter (im Flussdiagramm mit δ_T .. bezeichnet), on-line neue Werte eingeben, wie z.B. die Dauer der Mittelwertberechnung, das Umschaltintervall für die Analyse der beiden Gasquellen sowie das Entscheidungsintervall zur Regelung der Zuflussrate. Es bestand auch die Möglichkeit, jedes einzelne im Hintergrund laufende Programm mit Ausnahme vom "Datatransfer" zu stoppen oder neu zu starten.

"Kinetikvariablen"

Mit diesem Programm konnte der Benutzer alle Parameter zur Festlegung von pH-Grenzwerten und Korrektur der Substratzuflussrate (im Flussdiagramm mit δ_{pH} .. und δ_p .. bezeichnet) neu definieren, sowie während der pH-Elektrodeneichung die pH-Werterfassung stoppen. Dabei wurden die für die Regelung wichtigen und schon abgespeicherten pH-Werte entsprechend der Eichdifferenz neu angepasst, damit Fehlentscheidungen aufgrund der sprunghaftigen pH-Änderungen beim Eichen vermieden werden konnten.

4.3.2. Resultate mit dem Optimierungsprogramm

Da sich der Regelalgorithmus vor allem auf den gemessenen pH-Wert abstützte, war es wichtig, das Signal der pH-Sonde während einer längeren Zeitspanne zu untersuchen. Abbildung 20 zeigt den Verlauf des pH-Wertes im Filterüberstand während 145 Stunden. Die Durchmischung des Überstandes erfolgte lediglich durch die aufsteigenden Gasblasen. Das Signal verschob sich in dieser Zeitspanne um 0.1 pH-Einheiten nach unten und zwar nicht kontinuierlich, sondern sprunghaft. Aus den Signalschwankungen war ersichtlich, dass die minimal zu berücksichtigende pH-Änderung etwa 0.03 Einheiten betrug. Kleinere Änderungen verschwanden im Signalrauschen. Der Signalverlauf zeigte auch, dass ein häufiges Eichen der pH-Elektrode nötig war (ca. alle 70-80 Stunden), um Abweichungen des gemessenen vom tatsächlichen pH-Wert zu vermeiden. Diese Abweichungen hätten eine falsche Regelung der Substratzufuhr bewirkt, was in den meisten Fällen, bedingt durch die Struktur des Regelprogrammes, zu einer organischen Unterbelasung des Filters geführt hätte. Das Ziel der nächsten Versuche war, einen Zusammenhang zwischen der organischen Filterbelastung und der zeitlichen pH-Änderung darzustellen. Frühere Versuche hatten ergeben, dass sich bei organischen

PCS68000 HISTPLOT V.2.2 Process: 0 Oper: Etti
Name: AF Pumpcontrolled File: /h0/P00/d8604.27.1430
Start: 86/06/27 1859 Rec: 10 End: 86/07/03 1959 Rec: 300

86/10/09 1229

Time Scale: Start = 0

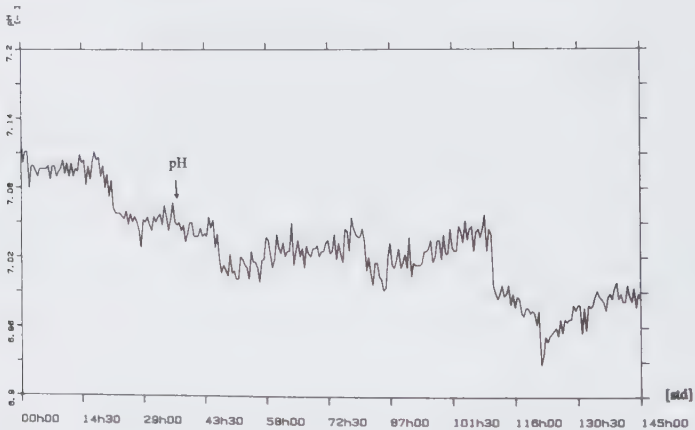


Abbildung 20: Zeitliche Änderung des ungemittelten pH-Signals.

Versuchsbedingungen:

Organische Filterbelastung L: $0.7 \text{ kg COD} \cdot \text{m}^{-3} \cdot \text{d}^{-1}$

Spezifische Methanproduktion P: $0.41 \text{ m}^3 \text{CH}_4 \cdot \text{m}^{-3} \cdot \text{d}^{-1}$

Verdünnungsrate D: $0.0045 \text{ Stunden}^{-1}$

Filterbelastungen bis zu $9 \text{ kg COD} \cdot \text{m}^{-3} \cdot \text{d}^{-1}$ im Filter selbstständig ein stabiler pH-Wert bei 7.1 einstellte (Abbildung 8). Abbildung 21 zeigt das Ergebnis eines Versuches zur Untersuchung der Geschwindigkeit von pH-Änderungen. Nach dem Einschalten der Substratzuflusspumpe stieg der pH-Wert zunächst an, möglicherweise wegen der sich durch den Substratfluss und die aufsteigenden Gasblasen bildenden Turbulenzen in der Flüssigkeit verursacht. Nach etwa einer Stunde begann eine Übersäuerung im Filter, worauf der pH-Wert zu sinken begann. Nach ca. 22 Stunden wurde die Substratzufuhr unterbrochen und die Zeit bis zum erneuten Anstieg des pH-Wertes gemessen. Diese in Tabelle 12 mit Reaktionsträgheit bezeichnete Zeit und die Anstiegsrate des pH-Wertes trugen zur Berechnung des Zeitfaktors T_{sleep} im Flussdiagramm (Abbildung 19) bei. Tabelle 12 fasst die Versuche bei den vier Belastungsstufen von 6 – $23 \text{ kg COD/m}^{-3} \cdot \text{d}^{-1}$ zusammen. Bei einer Belastung von $6 \text{ kg COD} \cdot \text{m}^{-3} \cdot \text{d}^{-1}$ war die pH-Abnahmerate, wie erwartet, vernachlässigbar, vermutlich verursacht durch die zunehmende Verschmutzung des Elektrodendiaphragmas. Bei Belastungen über $11 \text{ kg COD} \cdot \text{m}^{-3} \cdot \text{d}^{-1}$ stieg die pH-Abnahmerate nicht linear an. Die Dauer der Reaktionsträgheit stieg von 50 auf 180 Minuten. Interessant war die Tatsache, dass die pH-Anstiegsrate bei einer Belastung von $11 \text{ kg COD} \cdot \text{m}^{-3} \cdot \text{d}^{-1}$ grösser war als die Abnahmerate, dass bei höheren Belastungen jedoch die Anstiegsrate im Verhältnis zur Abnahmerate immer kleiner wurde.

L	pH-Abnahme	Reaktions- trägheit	pH-Zunahme
	beim Einschalten der Zuflusspumpe		beim Abschalten der Zuflusspumpe
$\text{kg COD} \cdot \text{m}^{-3} \cdot \text{d}^{-1}$	Stunden ⁻¹	Minuten	Stunden ⁻¹
6	0.0008	0	-0.0008
11	0.0109	50	0.0166
17	0.0237	60	0.0160
23	0.0391	180	0.0094

Tabelle 12: pH-Kinetik bei verschiedenen organischen Filterbelastungen L.

Bei den anschliessenden Versuchen mit dem Optimierungsprogramm erwies sich die Bestimmung des geeigneten Entscheidungsintervalls für die Korrektur der Pumpleistung und der anderen Regelparameter als besonders schwierig.

PCS68000 HISTPLOT V.2.2 Process: 0 Oper: Etti
Name: AF-PCP all data File: /h0/P00/d8606.10.1132
Start: 86/06/10 1433 Rec: 7 End: 86/06/12 0533 Rec: 85

86/10/09 1237

Time Scale: Start = 0

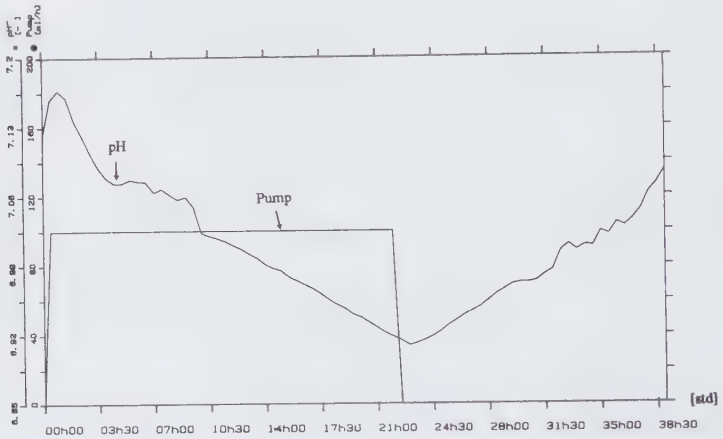


Abbildung 21: Versuch zur Bestimmung der Geschwindigkeit von pH-Änderungen.

Versuchsbedingungen:

Organische Filterbelastung L: $11 \text{ kg COD} \cdot \text{m}^{-3} \cdot \text{d}^{-1}$

Spezifische Methanproduktion P: $3.2 \text{ m}^3 \text{CH}_4 \cdot \text{m}^{-3} \cdot \text{d}^{-1}$

Verdünnungsrate D: $0.55 \text{ Stunden}^{-1}$

Abbildung 22 zeigt einen rechnergesteuerten Versuchsablauf mit den verwendeten Regelgrößen. Während des Anlaufprozesses betrug das Entscheidungsintervall 90 Minuten. Nach 6 Stunden Versuchsdauer, bei einer Pumpleistung von 130 ml/Stunde, erhöhte man das Entscheidungsintervall manuell auf 120 Minuten. Nach 23 Stunden fand eine Eichung der pH-Elektrode statt (E). Im stationären Zustand schwankte die Pumpleistung zwischen 110 und 130 ml/Stunde. Die Methanbildungsrate erreichte dabei Werte um 590 ml/Stunde, die Methankonzentration im gebildeten Gas betrug 77%. Umgerechnet ergab dies eine spezifische Methanproduktionsrate von maximal $1.3 \text{ m}^3 \text{CH}_4 \cdot \text{m}^{-3} \cdot \text{d}^{-1}$ bei einer organischen Belastung von $1.7\text{-}2.0 \text{ kg COD} \cdot \text{m}^{-3} \cdot \text{d}^{-1}$ und Retentionszeiten zwischen 100 und 84 Stunden. Die COD-Reduktion betrug bei Versuchsende 84%. Die erreichte maximale Filterbelastung lag deutlich unter jener mit "manueller Prozessregelung" (Abbildung 8, Tabelle 4).

Einige Versuche waren nötig, bis die Regelung der Substratzudosierung mit dem Optimierungsprogramm befriedigende Ergebnisse lieferte. In den meisten Fällen übersäuerte der Anaerobfilter und die Substratzufuhr wurde während Stunden gestoppt. Abbildung 23 zeigt einen Versuch, bei welchem die Regelung der Substratzuflussrate durch das Optimierungsprogramm ein zur manuellen Regelung vergleichbar gutes Ergebnis lieferte. Begonnen wurde mit einer Zuflussrate von 100 ml/Stunde (hydraulische Retentionszeit = 110 Stunden). In den folgenden 140 Stunden bewirkte das Regelprogramm die stufenweise Erhöhung der Durchflussrate auf einen Maximalwert von 420 ml/Stunde (hydraulische Retentionszeit = 26 Stunden). Um die Reaktion des Optimierungsprogrammes auf schnelle Erhöhungen der organischen Filterbelastung, verursacht durch einen gestiegenen COD-Gehalt des Schwemmwassers, verfolgen zu können, wurde durch Zugabe von Glucose der COD-Gehalt des Schwemmwassers von 7400 auf 8200 mg/l erhöht (in Abbildung 23 ist der Zeitpunkt der Zugabe durch Z markiert). Bei dieser erhöhten Filterbelastung begann der pH-Wert mit einer Rate von 0.014/Stunde zu sinken und bewirkte eine stufenweise Reduktion der Durchflussrate auf 370 ml/Stunde. Die Substratzuflussrate erreichte nach 155 Stunden einen stabilen Betrag um 380 ml/Stunde (hydraulische Retentionszeit = 29 Stunden). Im stationären Zustand betrug die spezifische Methanbildungsrate $3.9 \text{ m}^3 \text{CH}_4 \cdot \text{m}^{-3} \cdot \text{d}^{-1}$ bei einer organischen Belastung von $7.8 \text{ kg COD} \cdot \text{m}^{-3} \cdot \text{d}^{-1}$ und einer COD-Reduktion von 72%. Während des Versuchsablaufs wurde die pH-Elektrode zweimal geeicht (E), ohne dass dies einen Einfluss auf die Steuerung durch das Optimierungsprogramm gehabt hätte.

PCS68000 HISTPLOT V.2.2 Process: 0 Oper: Ett1
Name: PCP all data File: /h0/p00/vers4auto
Start: 86/06/25 1009 Rec: 1 End: 86/06/27 1209 Rec: 101

86/10/14 1201

Time Scale: Start = 0

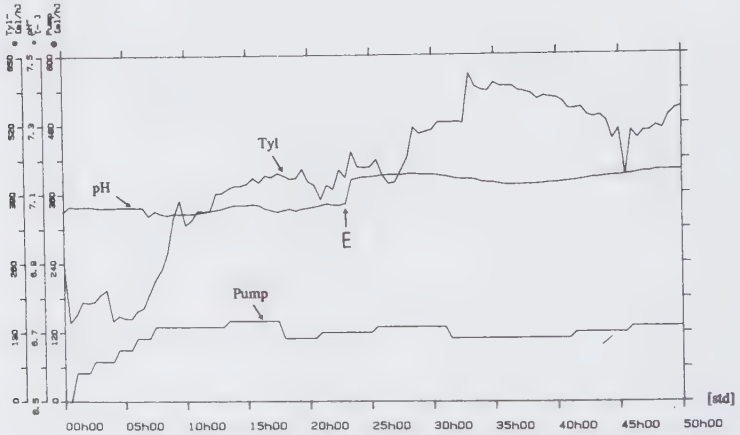


Abbildung 22: Prozessoptimierung mit einem Entscheidungsintervall von 120 Minuten.
Regelparameter:

δ_{Tint} :	120 Minuten
$\delta_{\Delta pH_{lim}}$:	$0.011 \text{ Stunde}^{-1}$
$\delta_{pH_{krit}}$:	6.8
$\delta_{pH_{rest}}$:	6.9
$\delta_{pH_{noise}}$:	0.04
$\delta_{P_{rest}}$:	$100 \text{ ml} \cdot \text{Stunde}^{-1}$
$\delta_{P_{1-6}}$:	10, 20, 30, -15, -20, -40 $\text{ml} \cdot \text{Stunde}^{-1}$
T_{sleep} :	50 Minuten

PCS68000 HISTPLOT V.2.2 Process: 0 Oper: Ett
Name: AF Pumpcontrolled File: /hd/p00/vare8auto
Start: 86/06/27 1429 Rec: 1 End: 86/07/04 0559 Rec: 320

86/10/14 1138

Time Scale: Start = 0

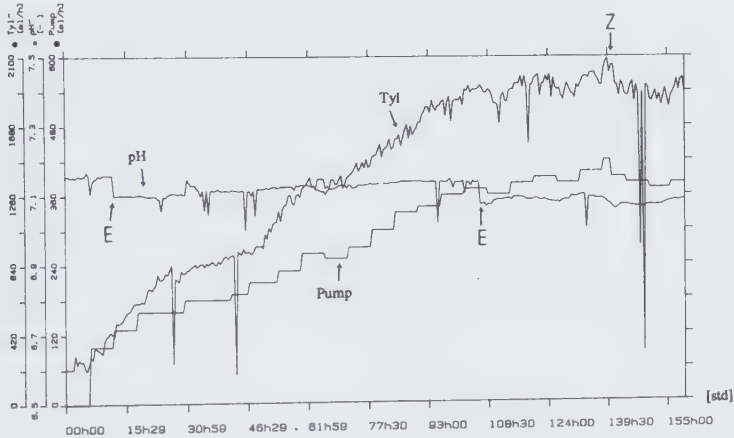


Abbildung 23: Prozessoptimierung mit einem Entscheidungsintervall von 360 Minuten.
Regelparameter:

δ_{Tint} :	360 Minuten
$\delta_{\Delta pHlim}$:	$0.011 \text{ Stunde}^{-1}$
δ_{pHkrit} :	6.8
δ_{pHrest} :	6.9
$\delta_{pHnoise}$:	0.02
δ_{Prest} :	$100 \text{ ml} \cdot \text{Stunde}^{-1}$
δ_{P1-6} :	10, 20, 30, -15, -20, -40 $\text{ml} \cdot \text{Stunde}^{-1}$
T_{sleep} :	50 Minuten

4.3.3. Diskussion der Optimierungsversuche

Die pH-statische Reaktionsführung mit einem empirischen Modell erreichte bei der Vergärung von Rübenschwemmwasser nicht ganz die Abbauleistungen der "manuellen" Prozessführung (Tabelle 4). Ein Grund war, dass man mit dem pH-Wert eine Messgrösse benutzte, die wohl über den physiologischen Zustand des Gesamtsystems eine Aussage machte, andererseits nicht genau ein Mass für die verschiedenen Abbau- und Umlagerungsvorgänge war. Wegen der grossen Pufferkapazität des Mediums im pH-Bereich von 6.5 bis 7.5 bewirkten kleine Änderungen der Konzentrationsverhältnisse an Fettsäuren nur geringe pH-Änderungen im Anaerobfilter. Um auch diese kleinen Konzentrationsänderungen zu erfassen, müssten pH-Schwankungen von 0.01 Einheiten noch erfasst werden können [43, 44], was bei der benützten Versuchsanordnung nicht möglich war. Beim Anaerobfilter fehlte auch eine Vorrichtung, die eine häufige automatische pH-Eichung ermöglichte, ohne dass die Elektrode hatte ausgebaut werden müssen.

Vergleiche mit anderen pH-statischen Prozessführungen mit komplexen Substraten an Anaerobfiltern sind in der Literatur kaum zu finden. Podruzni [45] benutzte die Gasaustrittsrate als Regelgrösse für sei Kontrollmodell an einem down-flow stationary fixed film reactor. Er wählte eine Proportional-Integral Regelung, mit welcher er eine maximale Gasaustrittsrate über eine längere Versuchsdauer konstant zu halten versuchte. Periodische Übersäuerungen und grosse Schwankungen in der Substratzuflussrate verhinderten jedoch eine optimale Prozessführung.

Um eine Verbesserung der Prozessführung zu erreichen, müsste das empirische Modell mindestens teilweise durch reaktionskinetische Modelle ersetzt werden. Versuche mit pH-statischer Prozessführung durch Messungen der CO_2 -Austrittsrate oder Berechnungen der Acetatkonzentration wurden schon in Fermentern mit einfachen Substratzusammensetzungen durchgeführt [46, 47].

Bei komplexen und wechselnden Substratzusammensetzungen kann die säurebildende von der methanbildenden Stufe getrennt werden, um die Kinetik der Umwandlung von Acetat zu Methan in gezielten Experimenten zu vermessen. Die Kenntnis dieser Kinetik ist Voraussetzung für eine mathematische Modellierung des Methangärprozesses, da siebzig Prozent des gebildeten Methans aus Acetat entstehen [48].

Für die Vergärung von Rübenschwemmwasser müsste diese räumliche Trennung der beiden Reaktionsstufen ebenfalls vorgenommen werden, um die Konzentrationen an Acetat und Propionat im Zulauf zum Anaerobfilter optimieren zu können.

5. Zusammenfassung

Zur Auswahl eines geeigneten Verfahrens für die Vergärung von Rübenschwemmwasser aus der Zuckerfabrik Frauenfeld wurden drei der bei Grossanlagen am häufigsten verwendeten Verfahren im Labormassstab miteinander verglichen: der anaerobe Kontaktprozess, der Anaerobfilter und der Anaerobfilter mit vorgeschalteter Versäuerungsstufe als zweistufiges Verfahren.

Der Anaerobfilter und das zweistufige Verfahren konnten mit einer organischen Filterbelastung von über $8 \text{ kg COD} \cdot \text{m}^{-3} \cdot \text{d}^{-1}$ bei einer COD-Abbauleistung von mindestens 70% betrieben werden. Beim Kontaktprozess hingegen betrug die organische Belastbarkeit bei vergleichbaren Abbauleistungen lediglich $4 \text{ kg COD} \cdot \text{m}^{-3} \cdot \text{d}^{-1}$.

Darauf wurde ein Anaerobfilter in der Zuckerfabrik Frauenfeld während drei Kampagnen als Pilotanlage eingesetzt. Die maximale organische Filterbelastung um eine COD-Abbauleistung von mindestens 70% zu erreichen, betrug bei der Pilotanlage $28 \text{ kg COD} \cdot \text{m}^{-3} \cdot \text{d}^{-1}$ mit einem Methan Ausbeutekoeffizienten von $0.34 \text{ m}^3 \text{CH}_4 / \text{kg COD}$. Diese gegenüber der Laboranlage erhöhte organische Belastbarkeit und die Fähigkeit, Schwankungen der Konzentration an organischen Substanzen im Schwemmwasser ohne Prozessstörungen zu überstehen, ergaben, dass eine Volumenvergrößerung der Laboranlage um einen Faktor 60 möglich war und sich der Einsatz eines Anaerobfilters auch unter Produktionsbedingungen bewährte.

Die schwankende Konzentration an organischen Substanzen im Schwemmwasser erfordert für eine optimale Abbauleistung des Anaerobfilters eine ständige Überwachung der Prozessdaten und eine Regelung der Schwemmwasserzufuhr. Ein Kleinrechner wurde deshalb zur Überwachung der Laboranlage eingesetzt. Das dazu entwickelte Kontrollprogramm entspricht einem empirischen Modell, welches einen Operator simuliert und in der Lage ist, den Einfluss von Konzentrationsschwankungen im Zulauf auszugleichen und den Vergärungsprozess bei einer optimalen organischen Belastung zu stabilisieren. Die durch das Kontrollprogramm erreichten Betriebsdaten entsprechen denjenigen bei permanenter Überwachung des Prozesses durch einen Operator.

6. Anhang

6.1. In der Arbeit verwendete Abkürzungen:

COD	Chemischer Sauerstoffbedarf ($\text{mg} \cdot \text{l}^{-1}$).
COD_{in}	COD-Konzentration im Zulauf ($\text{mg} \cdot \text{l}^{-1}$).
COD_{out}	COD-Konzentration im Überlauf ($\text{mg} \cdot \text{l}^{-1}$).
COD_r	COD-Abbauleistung (%).
D	Verdünnungsrate (Stunde^{-1}).
EAF	Eidg. Anst. für das forstliche Versuchswesen.
F	Substrat Umsatzrate ($\text{g} \cdot \text{l}^{-1} \cdot \text{Stunde}^{-1}$).
GVE	Grossvieheinheit.
HRT	hydraulische Retentionszeit (Stunden).
HRT_{min}	minimale hydraulische Retentionszeit.
K	Substratsättigungs Konstante.
L	organische Reaktorbelastung ($\text{kg COD} \cdot \text{m}^{-3} \cdot \text{d}^{-1}$).
M	Biomassekonzentration ($\text{g} \cdot \text{l}^{-1}$).
P	spezifische Methanproduktionsrate ($\text{m}^3 \text{CH}_4 \cdot \text{m}^{-3} \cdot \text{d}^{-1}$).
Pump	Pumpleistung Substratpumpe ($\text{ml} \cdot \text{Stunde}^{-1}$).
S, S_0	Substratkonzentration im Reaktor, bzw. im Substrat ($\text{g} \cdot \text{l}^{-1}$).
std	Stunden
Tyl	Methanbildungsrate ($\text{ml CH}_4 \cdot \text{Stunde}^{-1}$).
μ	spezifische Wachstumsrate (Stunde^{-1}).
μ_{max}	maximale spezifische Wachstumsrate (Stunde^{-1}).
V	Verweilzeit (Stunden).
Y	Methanausbeutekoeffizient ($\text{m}^3 \text{CH}_4 \cdot \text{kg COD}_{\text{in}}^{-1}$).
Y_0	maximaler Methanausbeutekoeffizient bei $\text{HRT} \rightarrow \infty$ ($\text{m}^3 \text{CH}_4 \cdot \text{kg COD}_{\text{in}}^{-1}$).

6.2 Herleitung der Methanbildungskinetik nach Chen [35]

In einem vollständig durchmischten System mit kontinuierlichem Zufluss werden die Änderung der Zellmasse und der Substratkonzentration durch folgende Differentialgleichungen beschrieben:

$$\frac{dM}{dt} = \mu M - \frac{M}{HRT} \quad (1)$$

$$\frac{dS}{dt} = -F + \frac{(S_o - S)}{HRT} \quad (2)$$

Der Zusammenhang von μ und F ist gegeben durch:

$$\mu = \frac{(Y/M)}{F} \quad (3)$$

Für den stationären Zustand gilt aus Gleichungen (1)-(3):

$$\mu = \frac{1}{HRT} \quad (4)$$

$$F = \frac{(S_o - S)}{HRT} \quad (5)$$

$$\frac{1}{Y} = \frac{(S_o - S)}{M} \quad (6)$$

Die Abhängigkeit der spezifischen Wachstumsrate von der Substratkonzentration lässt sich aus den Gleichungen (3)-(6) darstellen als:

$$\frac{\mu}{\mu_{max}} = \frac{S/S_o}{K + (1-K) S/S_o} \quad (7)$$

Aus der Gleichung (7) lässt sich die minimale Retentionszeit definieren als:

$$HRT_{min} = \frac{1}{\mu_{max}} \quad (8)$$

Für ein homogen durchmisches System im stationären Zustand ergeben die Gleichungen (4) und (7):

$$HRT = \frac{1}{\mu_{max}} + \frac{K (S_o - S)}{\mu_{max} S} \quad (9)$$

Die Konzentration des Substrates ist abhängig von der Substrat Eingangskonzentration:

$$\frac{S}{S_o} = \frac{K}{HRT / HRT_{min} - 1 + K} \quad (10)$$

Um die Kinetik der Vergärung komplexer Substrate zu beschreiben, wird anstelle der Substratkonzentration oft der COD-Wert des Substrates verwendet. Dabei wird nur derjenige Anteil berücksichtigt, welcher bei angenommener unendlich langer Retentionszeit noch abgebaut wird ($Y_o - Y$). Für Gleichung (10) gilt deshalb:

$$\frac{Y_o - Y}{Y_o} = \frac{K}{HRT / HRT_{min} - 1 + K} \quad (11)$$

oder:

$$Y = Y_o \left[1 - \frac{K}{HRT / HRT_{min} - 1 + K} \right] \quad (12)$$

Aus der Gleichung (12) ist ersichtlich, dass eine graphische Darstellung der Methanausbeute Y gegen $1/HRT$ die maximale theoretische Methanbildungsrate bei $HRT \rightarrow \infty$ ergibt.

Gleichung (11) lässt sich umschreiben als:

$$HRT = HRT_{min} + HRT_{min} K \left[\frac{Y}{Y_o - Y} \right] \quad (13)$$

Die Darstellung von HRT gegen $Y/(Y_o - Y)$ ergibt eine Gerade mit der Steigung $K \cdot HRT_{min}$ und dem Achsenabschnitt HRT_{min} .

6.3. Programme

```

/*      pumpdefs.c
      DEFINITIONEN
      purpose: constant definitions for PCP
      programmer: E. Da Pra          16. 8. 85
      change history: PCP-parameters added ed. 05. 06. 86
*/

/*
-----      channel level      -----
*/
#define PANIN_ST      0      /* first pump aichannel */
#define PANOUT_ST     4      /* first pump aochannel/derchannel */
#define PANIN         3      /* number of pump aichannels */
#define PANOUT        4      /* number of pump aochannels/derchannels */
#define C_PH          0      /* pH aichannel */
#define C_TYL         1      /* tylan aichannel */
#define C_HOR         2      /* horiba aichannel */
#define C_PUMP        3      /* pump aochannel */
#define C_PHM         4      /* pHmean derchannel */
#define C_TYLM        5      /* tylmean derchannel */
#define C_HORM        6      /* hormean derchannel */
#define C_CH4M        7      /* ch4flow derchannel */
#define C_CH4ALBIN    26     /* derchannel mean Horiba value for Albin */
#define C_GASUHRALBIN 25     /* aninput with gasflow from Wohlgroth */
#define C_CH4FLOWALBIN 27    /* derchannel ml CH4/h for Albin */
/*
-----      procs level      -----
*/
#define PUMPAD "/dd/PUMPDDESC/pptr" /* address of pump-data testfield */
#define PUMPPROCS      0      /* procs number for pump-control */
#define STOINTERV     1800    /* storage interval of PUMPPROCS */
#define CHANGINT      1800    /* valve-change interval [sec] */
#define WPOS_EDP       1      /* valve pos on edp-data */
#define WPOS_AD        0      /* valve pos on ad-data */
#define LOWPH         6.80    /* pH value to stop pump */
#define PHRESTART      6.90    /* pH value to restart pump */
#define PHNOISE        0.005  /* noise of pH-signal */
#define PHMAXVAL       20     /* arraysize of dpH dt values */
#define DEF_STRONGINCR 0.25    /* increment 25% of pump-performance */
#define DEF_INCR       0.10    /* " 10% of pump-performance */
#define DEF_LOWINCR    0.05    /* " 5% of pump-performance */
#define DEF_MAXDECR     0.4    /* decrement 40% of pump-performance */
#define DEF_DECR        0.2    /* " 20% of pump-performance */
#define DEF_MINDECR     0.1    /* " 10% of pump-performance */
#define HORL_DEF       1500    /* horiba lag-time [sec] */
#define HORINCR        1.0     /* minimal deltahoriba to be considererd */
#define PUMPL_DEF      14400    /* pump-change interval [sec] */
#define RESTARTPUMP_DEF 60.0    /* pumpvalue after program-restart */
#define MAXPUMP_DEF    200.0    /* upper pump-performance */
#define MINPUMP_DEF     0.0     /* minimal pump-performance */
#define MEANINTERV     900     /* mean values interval */
#define ACTIF          1
#define NOT_ACTIF      0

```

```

/*
----- file & program addresses -----
*/
#define TRANS      "/h0/PUMPRUN/pumptrans &"
#define PLANT      "/h0/PUMPRUN/pumplplant"
#define STATUS     "/h0/PUMPRUN/pumpstat"
#define KIN        "/h0/PUMPRUN/pumpkin"
#define MEAN       "/h0/PUMPRUN/pumpmean &"
#define GAS        "/h0/PUMPRUN/pumpgas &"
#define GASALBIN   "/h0/PUMPRUN/gasalbin &"
#define PUMP       "/h0/PUMPRUN/pumppcp"
#define ERROR      "/h0/PUMPRUN/pumpperror"
#define MASK1      "list /h0/PUMPRUN/pump1mask"
#define MASK2      "list /h0/PUMPRUN/pump2mask"
/*
----- I/O definitions -----
*/
#define SYSPRINTER  "/t1"
/*-----*/

/*
pumpglob.c
GLOBALE VARIABLEN
purpose: global definitions for pump-control program
programmer:   E. Da Pra   16. 6. 86
change history:

*/
typedef struct pp_param {
    float pumpval,      /* aoval for pump */
    pumpold,            /* old aoval for pump */
    deltapump,          /* pump-performance change [ml/h] */
    pumpupper,          /* upper pumplimit */
    pumplower,          /* lower pumplimit */
    pumprestart,        /* pumpvalue after restart */
    stronginc,          /* value for strong pumpincrement */
    incr,               /* " " normal " */
    lowincr,            /* " " low " */
    strongdec,          /* " " strong pumpdecrement */
    decr,               /* " " normal " */
    lowdecr,            /* " " low " */
    pHin,               /* pH aival */
    tylin,              /* tylin aival */
    horin,              /* horiba aival */
    pHmean,             /* mean pH-value */
    pHoldval,           /* old pH reference-value for AF-PCP */
    horoldval,          /* " hor " " " " */
    tyldval,            /* " tyl " " " " */
    pHnewval,           /* new pH " " " " */
    hornval,            /* " hor " " " " */
    tylnval,            /* " tyl " " " " */
    deltaph,            /* change of pH value between pumpcontrols */
    deltahor,           /* " " hor " " " */
    deltatylin,         /* " " tyl " " " */
    dpHold,             /* last dpH */
    dpHdt[PHMAXVAL],    /* array of dpH t act values */
    dpH_t_max,          /* pH-decrease value to stop pump */
    dpH_t_act,          /* actual pH decrease [/h] */
    pH_low,             /* pH-value to stop pump */
    pH_restart,         /* pH-value to restart pump */
    tylmean,            /* mean tylin-value */
    hormean,            /* mean horiba-value */
    ch4flow;            /* mean CH4-flow value */
};

```

```

int    horilag,          /* horiba lag-time after valve change */
      horint,           /* horiba change interval */
      hornext,          /* next Horiba change in julsecs */
      nexthormean,      /* time until next mean values */
      untilhoriba,      /* time until next integration */
      record,           /* number of record stored */
      nextsto,          /* next storage time for PUMPPROCS */
      changint,         /* pump-adjust intervall */
      nextchange,        /* next pump-adjust event */
      untilnext,        /* time until next pump-adjust */
      controlstart,     /* start of AF-PCP with getsecs() */
      meanint,          /* interval of mean-values calculation */
      nextmean,         /* next mean values in secs */
      pHcount,          /* number of values in dpHdt[] array */
      pHfull,           /* dpHdt[]-array full */
      slopeflag,        /* evaluates dpHdt[]-array */
      stopflag,         /* Automatic stop(=1) */
      pumpsleep,        /* pumpchange active=0 inactive=1 */
      calibflag,        /* calibration of pH-electrode 0=- 1=+ */
      plstatus,         /* graphic display (ACTIF or NOT_ACTIF) */
      pidplant,         /* pid of forked "pumplant" */
      pidprint,         /* pid of forked "pumperror" */
      transflag,
      meanflag,
      gasflag,
      pumpflag,
      albinflag,
      codin,            /* COD_in value [mgO2/l] */
      codout,           /* COD_out value [mgO2/l] */
      wpos;            /* valve position (0=ed,1=ad) */
char   } PAN, *PANPTR;

/*-----*/

/* pumptrans.c
   DATATRANSFER
   purpose:   Sets initialisation values for PCP
              makes directory /dd/PUMPDESC
              The program is started by 'pumpload'.
   programmer:                                     ed. 16. 06. 86.
   change history:  calib-flag & program-flags
                   added                                     ed. 08. 07. 86.
                   errors printed from forked
                   program 'pumperror'                       ed 17. 07. 86.

*/

#include "pumpdefs.c"
#include "pumpglob.c"
#include <pcs.h>
static direct PAN ps;
static direct PANPTR pumpfix;

main () {
    register PANPTR a;
    register int i;
    unsigned ptr, channel;
    int c, pn, pf;
    pf=fopen(SYSPRINTER, "w");
    if ((getprstat(PUMPPROCS) != 1) || (getstostat(PUMPPROCS) < 1)) {

```

```

fprintf (pf, "*****\n");
fprintf (pf, "      First activate Process Nr.%d      *\n", PUMPPROCS);
fprintf (pf, "      Then set storage interval to 30min  *\n");
fprintf (pf, "*****\n");
exit (0);
}
system ("mkdir /dd/PUMPDESC");
a = pumpfix = & ps;
fprintf (pf, "mkdir /dd/PUMPDESC completed\n");
if ((pn = creat (PUMPAD, 2)) == -1) {
    fprintf (pf, "ERROR! <pumptrans> cannot create PUMPAD\n");
    exit (0);
}
ptr = pumpfix;
write (pn, &ptr, sizeof (unsigned));
close (pn);
fprintf (pf, "pumptrans & successfully started! ptr=%d\n", ptr);
if ((a->pidprint=os9fork (ERROR, 0, 0, 1, 1, 0, 128)) == -1)
    fprintf (pf, "ERROR! Fork to <pumperror> failed!\n");
sleep (10);
fclose (pf);
/*----- initialisation -----*/
a->pumpval      = 0.0;
a->dpH_t_max    = 0.0;
a->dpH_t_act    = 0.0;
a->pH_low       = LOWPH;
a->pH_restart   = PHRESTART;
a->untilhoriba = 0;
a->horlag       = HORL_DEF;
a->changint     = PUMPL_DEF;
a->meanint      = MEANINTERV;
a->horint       = CHANGINT;
a->pHmean       = 0.0;
a->tylmean      = 0.0;
a->hormean      = 0.0;
a->ch4flow      = 0.0;
a->codin        = 0.0;
a->codout       = 0.0;
a->stronginc    = DEF_STRONGINCR;
a->lowinc       = DEF_LOWINCR;
a->incr         = DEF_INCR;
a->strongdec    = DEF_MAXDECR;
a->lowdec       = DEF_DECR;
a->decr         = DEF_MINDECR;
a->plstatus     = NOT_ACTIF;
a->pumpupper    = MAXPUMP_DEF;
a->pumplower    = MINPUMP_DEF;
a->pumprestart  = RESTARTPUMP_DEF;
a->calibflag    = 0;
a->stopflag     = 0;
a->transflag    = ACTIF;
a->meanflag     = NOT_ACTIF;
a->gasflag      = NOT_ACTIF;
a->pumpflag     = NOT_ACTIF;
a->albinflag    = NOT_ACTIF;

```

```

/*----- main loop -----*/
for (;;) {
    if (a->calibflag == 0) {
        for (channel = PANIN_ST, a=pumpfix, i=0; i<=PANIN; channel++, i++) {
            if (channel == C_PH) a->pHin = getan(channel);
            if (channel == C_TYL) a->tylin = getan(channel);
            if (channel == C_HOR) a->horin = getan(channel);
        }
        a->nextsto = getstotim(PUMPPROCS)-getsecs();
        a->record = getrecords(PUMPPROCS);
        for (channel = PANOUT_ST, a=pumpfix, i=0; i<=PANOUT; channel++, i++) {
            if (channel == C_PHM) setder(channel, a->pHmean);
            if (channel == C_TYLM) setder(channel, a->tylmean);
            if (channel == C_HORM) setder(channel, a->hormean);
            if (channel == C_CH4M) setder(channel, a->ch4flow);
        }
        if (a->stopflag == 1) {
            if ((getprstat(PUMPPROCS) != 1) || (getstostat(PUMPPROCS) < 1)) {
                kill(a->pidprint, 1000);
                sleep(5);
                kill(a->pidprint, 2);
                sleep(2);
                a->meanflag = NOT_ACTIF;
                a->gasflag = NOT_ACTIF;
                a->pumpflag = NOT_ACTIF;
                sleep(10);
                exit (0);
            }
        }
        if (a->transflag == NOT_ACTIF) {
            a->meanflag = NOT_ACTIF;
            a->gasflag = NOT_ACTIF;
            a->pumpflag = NOT_ACTIF;
            kill(a->pidprint, 1001);
            sleep(5);
            exit (0);
        }
        sleep (1);
    } /* for (;;) */
} /* main */
/*-----*/

```

```

/*      pumperror.c
        FEHLERMELDUNGEN fuer DRUCKER
        purpose:  prints all error-statements from the
                  background tasks: pumptrans&
                           pumpmean&
                           pumpgas&
                           pumppcp&

        programmer:                                ed.  07. 14. 86.

        change history:

*/

#include "pumpdefs.c"
#include "pumpglob.c"
#include <pcs.h>
static direct PANPTR pumpfix;
static direct TIPTTR timfix;
static direct TI ts;
int code,pf;

main() {
    int seconds,min,hour,pa,intrupt();
    register PANPTR a;
    register TIPTTR b;
    unsigned ptr;

    intercept (intrupt);
    pf=fopen(SYSPRINTER,"w");
    if ((pa=open(PUMPAD,1)) == -1) {
        fprintf(pf,"** ERROR! <pumperror> cannot open 'PUMPAD'!\n");
        exit(0);
    }
    read(pa,&ptr,sizeof(unsigned));
    a = pumpfix = ptr;
    close(pa);
    fprintf(pf,"* pumperror successfully started ptr=%d\n",ptr);
    b = timfix = &ts;
    for(;;) {
        tsleep(0);
        switch(code) {
            case 1000:
                fprintf (pf,"*****\n");
                fprintf (pf,"*          First activate Process Nr.%d          *\n",PUMPPROCS);
                fprintf (pf,"*          Then set storage interval to 30min          *\n");
                fprintf (pf,"*****\n");
                printim(b);
                break;
            case 1001:
                fprintf (pf,"*****\n");
                fprintf (pf,"* <pumptrans> stopped by operator!          *\n");
                fprintf (pf,"* All background programs stopped!          *\n");
                fprintf (pf,"*****\n");
                printim(b);
                break;

```

```
case 1002:
fprintf (pf, "*****\n");
fprintf (pf, "** <pumpmean> stopped by operator!      *\n");
fprintf (pf, "*****\n");
printim(b);
break;
case 1003:
fprintf (pf, "*****\n");
fprintf (pf, "** <pumpgas> stopped by operator!      *\n");
fprintf (pf, "*****\n");
printim(b);
break;
case 1004:
fprintf (pf, "*****\n");
fprintf (pf, "** <gasalbin> stopped by operator!      *\n");
fprintf (pf, "*****\n");
printim(b);
break;
case 1005:
fprintf (pf, "*****\n");
fprintf (pf, "** <pumppcp> stopped by operator!      *\n");
fprintf (pf, "*****\n");
printim(b);
break;
case 1010:
    seconds = getsecs()-a->controlstart;
    hour = abs(seconds/3600);
    min = abs((seconds-(hour*3600))/60);
fprintf (pf, "*****\n");
fprintf (pf, "** AF-PCP 68000 Age %04d:%02d hours      *\n",hour,min);
fprintf (pf, "** delta pH :%1.5f      *\n",a->deltapH);
fprintf (pf, "** delta Horiba :%02.4f      *\n",a->deltahor);
fprintf (pf, "** delta CH4flow :%02.4f      *\n",a->deltatyl);
fprintf (pf, "** -----*\n");
fprintf (pf, "** >> pumpchange :%03.4f ml/h      *\n",a->deltapump);
fprintf (pf, "** new pumpvalue :%03.4f ml/h      *\n",a->pumpval);
fprintf (pf, "*****\n");
printim(b);
break;
case 1011:
    seconds = getsecs()-a->controlstart;
    hour = abs(seconds/3600);
    min = abs((seconds-(hour*3600))/60);
fprintf (pf, "*****\n");
fprintf (pf, "** AF-PCP 68000 Age %04d:%02d hours      *\n",hour,min);
fprintf (pf, "** pH-value at critical level!      *\n");
fprintf (pf, "** Pumpperformance set to 0.0 ml/h      *\n");
fprintf (pf, "** Waiting until pH is:%1.5f      *\n",a->pH_restart);
fprintf (pf, "*****\n");
printim(b);
break;
```

```

case 1012:
    seconds = getsecs()-a->controlstart;
    hour = abs(seconds/3600);
    min = abs((seconds-(hour*3600))/60);
    fprintf(pf, "*****\n");
    fprintf(pf, " AF-PCP 68000 Age %04d:%02d hours *\\n", hour, min);
    fprintf(pf, " pH-value is: %1.5f *\\n", a->pHmean);
    fprintf(pf, " AF-PCP restarted! *\\n");
    fprintf(pf, "*****\n");
    printim(b);
    break;
}
} /* for(;;) */
fclose(pf);
} /* main */
/*----- printim -----*/
printim (b)
{
    register TIPTTR b;
    {
        getdate(getsecs(),b);
        fprintf(pf, "Date of message: ");
        fprintf(pf, "%2d/%02d/%02d ", b->year, b->month, b->day);
        fprintf(pf, " %02d%02d:%02d\\n", b->hour, b->min, b->sec);
        fprintf(pf, "\\n");
        fprintf(pf, "\\n");
        fprintf(pf, "\\n");
        fprintf(pf, "\\n");
    }
}
/*----- intrupt -----*/
intrupt(sig) {
    code = sig;
    if (sig == 2) exit(0);
}
/*-----*/

/*      pumpmean.c
        MITTELWERTE
        purpose: calculates pH_mean & tytan_mean
        programmer: E. Da Pra                                24. 04. 86.

        change history: with interface routines                ed      15. 05. 86.
                        with new ch4flow calculation            ed      05. 07. 86.
*/

#include "pumpdefs.c"
#include "pumpglob.c"
#include <pcs.h>
static direct PANPTR pumpfix;

main () {
    register PANPTR a;
    register int count, shifter;
    unsigned ptr;
    int c, loop, nextcalc, calcbeg, calcend, deltetime, pn, pf;
    float pH, tyl, time, phor, fact;

```

```
/* pointer initialisation */
pf = fopen(SYSPRINTER,"w");
if ((pn = open (PUMPAD,1)) == -1) {
    fprintf(pf,"* ERROR! <pumpmean> cannot open PUMPAD! *\n");
    fclose(pf);
    exit(0);
}
read (pn,&ptr,sizeof(unsigned));
close (pn);
fprintf(pf,"* pumpmean & successfully started! ptr=%d\n",ptr);
fclose(pf);
a = pumpfix = ptr;
a->meanflag = ACTIF;
a->pHcount = 0;
a->pHfull = 0;
loop = 0;
/* main loop */
for (;;) {
    pH = a->pHin;
    tyl = a->tylin;
    nextcalc = getsecs()+a->meanint;
    for (a=pumpfix,count=2;nextcalc-getsecs() >=7;count++) {
        a->nextmean = nextcalc-getsecs();
        pH = pH+a->pHin;
        tyl = tyl + a->tylin;
        if (a->meanflag == NOT_ACTIF) {
            kill(a->pidprint,1002);
            sleep(5);
            exit(0);
        }
        sleep (1);
    }
    a->pHmean = pH/(count-1);
    a->tylmean = tyl/(count-1);
    phor = a->hormean/100;
    fact = (phor*0.0645+0.3111)/(0.3106*(0.941-0.061*phor));
    a->ch4flow = a->tylmean*phor*fact;
    setder(C_CH4M,a->ch4flow);
    if (loop == 1) {
        calcend = getsecs();
        time = deltetime = calcend-calcbeg;
        a->dpH_t_act=((a->pHmean-a->dpHold)/time)*3600.0;
        calcbeg = calcend;
        a->dpHold = a->pHmean;
        if(a->pHfull == 1) {
            for(shift=1;shift<=19;shifter++) {
                a->dpHdt[shifter-1] = a->dpHdt[shifter];
            }
            a->dpHdt[a->pHcount] = a->dpH_t_act;
        }
        if(a->pHfull == 0) {
            a->dpHdt[a->pHcount] = a->dpH_t_act;
            a->pHcount++;
            if(a->pHcount == 20) {
                a->pHfull = 1;
                a->pHcount--;
            }
        }
    }
}
```

```

if (loop == 0) {
    calcbeg = getsecs();
    a->dpHold = a->pHmean;
    loop = 1;
}
sleep (7);
} /* for(;;) */
} /* main */
/*-----*/
/*
    pumpgas.c
    GASANALYSE
    purpose: control of gas change system and storage of values
             switch-time depends on storage interval of process 0
             digital outputs VAL1,VAL2 and MANUAL afflicted

    programmer: E. Da Pra          30. 4. 86

    change history: uses interface functions          15. 05. 86
                   calculations independent of
                   storage interval of PROC 0        13. 06. 86
                   corrections for gasflow added      13. 06. 86
*/

#include "pumpdefs.c"
#include "pumpglob.c"
#include <pcs.h>
static direct PANPTR pumpfix;

main () {
    register PANPTR a;
    register int count;
    int c,pn,pf,starttime;
    float ch4,horiba;
    unsigned ptr;

/* pointer initialisation */
pf =fopen(SYSPRINTER,"w");
if ((pn = open (PUMPAD,1)) == -1) {
    fprintf(pf,"* ERROR! <pumpgas> cannot open PUMPAD! *\n");
    exit(0);
}
read (pn,&ptr,sizeof(unsigned));
a = pumpfix = ptr;
close (pn);
fprintf(pf,"* pumpgas & successfully started! ptr=%d\n",ptr);
fclose(pf);

/* Gas change system initialisation */
a->gasflag = ACTIF;
setdi(2,1);
setder(C_CH4ALBIN,0.0);
a->wpos = WPOS_EDP;
changew ();

```

```

/* main loop */
for (a=pumpfix;;) {
    a->hornext = getsecs()+a->horlag;
    count = 1;
    for (starttime = getsecs(); (starttime+a->horint-getsecs()) >=0;) {
        a->nexthormean = (starttime+a->horint-getsecs());
        if ((a->untilhoriba = a->hornext-getsecs()) <=0) {
            if (count == 1) ch4 = a->horin;
            else {
                ch4 = ch4+a->horin;
                count++;
            }
            sleep(1);
        }
        if (a->gasflag == NOT_ACTIVIF) {
            kill(a->pidprint,1003);
            sleep(5);
            exit(0);
        }
        sleep(3);
    }
    horiba = ch4/count;
    switch (a->wpos) {
        case 0 : setder(C_CH4ALBIN,horiba);
                changew ();
                sleep (3);
                break;
        case 1 : a->hormean = horiba;
                changew ();
                sleep (3);
                break;
    }
} /* for (;;) */
} /* main */

/*----- changew -----*/
changew () {
    switch (pumpfix->wpos) {
        case 0: setdi(0,1);
                setdi(1,0);
                pumpfix->wpos = WPOS_EDP;
                break;
        case 1: setdi(0,0);
                setdi(1,1);
                pumpfix->wpos = WPOS_AD;
                break;
    }
}
/*-----*/

```

```
/*      pumppcp.c
        PROZESS OPTIMIERUNG
        purpose: pump-control program
        programmer: E. Da Pra          24. 4. 86

        change history: with interface routines          ed. 15. 05. 86.
*/

#include "pumpdefs.c"
#include "pumpglob.c"
#include <pcs.h>
static direct PANPTR pumpfix;
int startpump,pf;

main () {
    register PANPTR a;
    unsigned ptr;
    int pn,
        mode = 0;

/* pointer initialisation */
    pf = fopen(SYSPRINTER,"w");
    if ((pn = open (PUMPAD,1)) == -1) {
        fprintf(pf,"* ERROR! <pumppcp> cannot open PUMPAD! *\n");
        exit(0);
    }
    read (pn,&ptr,sizeof(unsigned));
    close (pn);
    a = pumpfix = ptr;
    a->pumpsleep = 0;
    startpump = 0;
    fprintf(pf,"AF PCP PROGRAM has started   ptr=%d\n",ptr);
    a->controlstart = getsecs();
    a->nextchange = getsecs()+a->changint;
    fprintf(pf,"start=%d nextchange=%d\n",a->controlstart,a->nextchange);
    fclose(pf);
    setout(C PUMP,a->pumpval);
    a->pumpold = a->pumpval;
/* main loop */
    for (a=pumpfix;;) {
        if (startpump == 0) {
            if (((a->controlstart+a->meanint)-getsecs()) < 0) {
                a->pHoldval = a->pHmean;
                a->horoldval = a->hormean;
                a->tyloldval = a->ch4flow;
                startpump = 1;
            }
        }
        if (a->pumpsleep = 0) {
            if ((a->untilnext = (a->nextchange-getsecs())) < 0) {
                a->pHnewval = a->pHmean;
                a->hornewval = a->hormean;
                a->tylnewval = a->ch4flow;
                a->deltapH = a->pHnewval - a->pHoldval;
                a->deltahor = a->hornewval - a->horoldval;
            }
        }
    }
}
```

```
a->deltatyl = a->tylnewval - a->tyloldval;
a->pHoldval = a->pHnewval;
a->horoldval = a->hornewval;
a->tyloldval = a->tylnewval;
if (a->deltapH > PHNOISE) {
    if (a->deltahor >= HORINCR) mode = 1;
    else mode = 2;
}
if (a->deltapH <= -PHNOISE) {
    if ((a->deltahor >= 0) && (a->deltatyl >= 0)) mode = 5;
    if ((a->deltahor <= 0) || (a->deltatyl <= 0)) mode = 4;
}
if ((a->deltapH >= -PHNOISE) && (a->deltapH <= PHNOISE)) {
    if ((a->deltatyl > 0.0) && (a->deltahor > 0.0)) mode = 6;
    if ((a->deltatyl < 0.0) && (a->deltahor < 0.0)) mode = 3;
}
pumpchange(a,mode);
a->nextchange = getsecs()+a->changint;
mode = 0;
} /* if a->untilnext < 0 */
if (a->pumpold != a->pumpval) {
    setout(C PUMP,a->pumpval);
    a->pumpold = a->pumpval;
}
} /* if a->pumpsleep == 0 */
if (a->pHmean <= a->pH_low) {
    setout(C PUMP,0.0);
    a->pumpold = a->pumpval = 0.0;
    a->pumpsleep = 1;
    a->nextchange = -111;
    kill(a->pidprint,1011);
    sleep(4);
}
if (a->pumpsleep == 1) {
    if (a->pHmean >= a->pH_restart) {
        setout(C PUMP,a->pumprestart);
        a->pumpold = a->pumpval = a->pumprestart;
        a->pumpsleep = 0;
        startpump = 0;
        a->nextchange = getsecs()+a->changint;
        kill(a->pidprint,1012);
        sleep(4);
    }
}
sleep(10);
} /* for(;;) */
} /* main */
```

```
/*----- pumpchange -----*/
pumpchange (a,mode)
PANPTR a;
int mode;
{
    switch (mode) {
        case 0: a->deltapump = 0.0;
            break;
        case 1: a->deltapump = a->pumpval*a->stronginc;
            a->pumpval = a->pumpval+a->deltapump;
            break;
        case 2: a->deltapump = a->pumpval*a->incr;
            a->pumpval = a->pumpval+a->deltapump;
            break;
        case 3: a->deltapump = a->pumpval*a->lowincr;
            a->pumpval = a->pumpval+a->deltapump;
            break;
        case 4: a->deltapump = -(a->pumpval*a->strongdec);
            a->pumpval = a->pumpval+a->deltapump;
            break;
        case 5: a->deltapump = -(a->pumpval*a->decr);
            a->pumpval = a->pumpval+a->deltapump;
            break;
        case 6: a->deltapump = -(a->pumpval*a->lowdecr);
            a->pumpval = a->pumpval+a->deltapump;
            break;
    }
    if (a->pumpval < 0.0) a->pumpval = 0.0;
    if (a->pumpval > a->pumpupper) a->pumpval=a->pumpupper;
    if (a->pumpval < a->pumplower) a->pumpval=a->pumplower;
    setout(C_PUMP,a->pumpval);
    a->pumpold = a->pumpval;
    kill(a->pidprint,1010);
    sleep(2);
}
/*-----*/
```

```
/*      pumpkin.c
INTERAKTIVES PROGRAMM: KINETIKVARIABLEN
purpose :   display and change of kinetic-parameters
programmer:                               ed. 26/06/86
change history:

*/

#include "pumpdefs.c"
#include "pumpglob.c"
#include "monaux.c"
#include "errmes.c"
#include <pcs.h>

static direct PANPTR pumpfix;
static direct TIPTR timfix;
static direct TI ts;
int sigflag;

main () {
    char *inbu;
    int inrupt(),pa;
    unsigned ptr;
    double atof();

    timfix = &ts;
    intercept(inrupt);
    system ("tmode nopause");
    clrscr();
    backinvers();
    system (MASK2);
    inversfor();
    if ((pa=open(PUMPAD,1)) == -1) {
        clmess();
        write (0,mess1,sizeof(mess1));
        alarm();
        exit (0);
    }
    read (pa,&ptr,sizeof(unsigned));
    pumpfix = ptr;
    close (pa);
    sigflag = 0;
    for (;;) {
        curback();
        readln (0,inbu,2);
        if ((*inbu != 'c') && (*inbu != 'd')) {
            clmess ();
            write (0,mess2,sizeof(mess2));
            sleep(4);
            clmess ();
            curback ();
        }
        if (*inbu == 'd') display (pumpfix,timfix);
        if (*inbu == 'c') change (pumpfix,timfix);
    }
}
```

```
/*----- display -----*/
display (a,t)
    register PANPTR a;
    register TIPTR t;
{
    int secs,minute,time;

    for (;;) {
        if (sigflag == 1) break;
        clffield (2,1,48); printf("%d\n",PUMPPROCS);
        clffield (2,3,13); printf("%d\n",getprstat(PUMPPROCS));
        clffield (4,3,32); printf("%d\n",a->record);
        time = a->nextsto;
        minute = abs(time/60);
        secs = time - (minute*60);
        clffield (6,3,60); printf("%02d:%02d\n",minute,secs);
        clffield (8,6,31); printf("%4.1f\n",a->pumpval);
        clffield (8,6,67); printf("%4.1f\n",a->pumprestart);
        clffield (8,7,31); printf("%4.1f\n",a->pumpupper);
        clffield (8,7,67); printf("%4.1f\n",a->pumplower);
        clffield (8,8,31); printf("%2.3f\n",a->stronginc*100);
        clffield (8,8,67); printf("%2.3f\n",a->incr*100);
        clffield (8,9,31); printf("%2.3f\n",a->lowincr*100);
        clffield (8,9,67); printf("%2.3f\n",a->strongdec*100);
        clffield (8,10,31); printf("%2.3f\n",a->decr*100);
        clffield (8,10,67); printf("%2.3f\n",a->lowdecr*100);
        clffield (8,12,31); printf("%1.4f\n",a->pH_low);
        clffield (8,12,67); printf("%1.4f\n",a->pH_restart);
        clffield (8,13,31); printf("%1.4f\n",a->dpH_t_act);
        clffield (8,13,67); printf("%1.4f\n",a->dpH_t_max);
        clffield (8,15,31);
        if (a->calibflag == 0) printf("OFF\n");
        else printf("ON\n");
        clffield (8,15,67);
        if (a->stopflag == 0) printf("OFF\n");
        else printf("ON\n");
        distim (t);
        tsleep (150);
    }
    sigflag = 0;
}
/*----- change -----*/
change (a,t)
    register PANPTR a;
    register TIPTR t;
{
    char inbu[6];
    int secs,minute,time;

    for (;;) {
        if (sigflag == 1) break;
        clffield (2,1,48); printf("%d\n",PUMPPROCS);
        clffield (2,3,13); printf("%d\n",getprstat(PUMPPROCS));
        clffield (4,3,32); printf("%d\n",a->record);
        time = a->nextsto;
        minute = abs(time/60);
        secs = time - (minute*60);
```

```
clfield (6,3,60); printf("%02d:%02d\n",minute,secs);
clfield (8,6,31); printf("%4.1f\n",a->pumpval);
clfield (8,6,67); printf("%4.1f\n",a->pumprestart);
clfield (8,7,31); printf("%4.1f\n",a->pumpupper);
clfield (8,7,67); printf("%4.1f\n",a->pumplower);
clfield (8,8,31); printf("%2.3f\n",a->stronginc*100);
clfield (8,8,67); printf("%2.3f\n",a->incr*100);
clfield (8,9,31); printf("%2.3f\n",a->lowincr*100);
clfield (8,9,67); printf("%2.3f\n",a->strongdec*100);
clfield (8,10,31); printf("%2.3f\n",a->decr*100);
clfield (8,10,67); printf("%2.3f\n",a->lowdecr*100);
clfield (8,12,31); printf("%1.4f\n",a->pH_low);
clfield (8,12,67); printf("%1.4f\n",a->pH_restart);
clfield (8,13,31); printf("%1.4f\n",a->dpH_t_act);
clfield (8,13,67); printf("%1.4f\n",a->dpH_t_max);
clfield (8,15,31);
if (a->calibflag == 0) printf("OFF\n");
else printf("ON\n");
clfield (8,15,67);
if (a->stopflag == 0) printf("OFF\n");
else printf("ON\n");
distim(t);
clfield (2,18,42);
readln (0,inbu,2);
switch (*inbu) {
    case 'a': clfield (8,6,31);
               readln (0,inbu,6);
               a->pumpval = atof(inbu);
               break;
    case 'b': clfield (8,6,67);
               readln (0,inbu,6);
               a->pumprestart = atof(inbu);
               break;
    case 'c': clfield (8,7,31);
               readln (0,inbu,6);
               a->pumpupper = atof(inbu);
               break;
    case 'd': clfield (8,7,67);
               readln (0,inbu,6);
               a->pumplower = atof(inbu);
               break;
    case 'e': clfield (8,8,31);
               readln (0,inbu,6);
               a->stronginc = atof(inbu)/100;
               break;
    case 'f': clfield (8,8,67);
               readln (0,inbu,6);
               a->incr = atof(inbu)/100;
               break;
    case 'g': clfield (8,9,31);
               readln (0,inbu,6);
               a->lowincr = atof(inbu)/100;
               break;
    case 'h': clfield (8,9,67);
               readln (0,inbu,6);
               a->strongdec = atof(inbu);
               break;
    case 'i': clfield (8,10,31);
               readln (0,inbu,6);
               a->decr = atof(inbu)/100;
               break;
```

```
case 'k': clfield (8,10,67);
    readln (0,inbu,6);
    a->lowdecr = atof(inbu)/100;
    break;
case 'l': clfield (8,12,31);
    readln (0,inbu,6);
    a->pH_low = atof(inbu);
    break;
case 'm': clfield (8,12,67);
    readln (0,inbu,6);
    a->pH_restart = atof(inbu);
    break;
case 'n': clfield (8,13,31);
    printf("ONLINE\n");
    break;
case 'o': clfield (8,13,67);
    readln (0,inbu,6);
    a->dpH_t_max = atof(inbu);
    break;
case 'p': clfield (8,15,31);
    if (a->calibflag == 0) {
        a->calibflag = 1;
        printf("ON\n");
    }
    if (a->calibflag == 1) {
        a->calibflag = 0;
        printf("OFF\n");
    }
    break;
case 'q': clfield (8,15,67);
    if (a->stopflag == 0) {
        a->stopflag = 1;
        printf("ON\n");
    }
    if (a->stopflag == 1) {
        a->stopflag = 0;
        printf("OFF\n");
    }
    break;
}
inbu[0]='0';
} /* for(;;) */
sigflag = 0;}
/*----- distim -----*/
distim (t)
    register TIPTR t; {
    getdate (getsecs(),t);
    clfield (17,1,57);
    printf ("%2d/%02d/%02d ",t->year,t->month,t->day);
    printf (" %02d%02d:%02d\n",t->hour,t->min,t->sec);
}
/*----- intrupt -----*/
intrupt (sig) {
    if (sig == 2) {
        clrscrho();
        backinvers();
        exit(0);
    }
    if (sig == 3) sigflag = 1;
}
```

```
/*      pumpstat.c
      INTERAKTIVES PROGRAMM: ZEITVARIABLEN
      purpose :   display and change of control-parameters
                  start of graphics
      programmer:
      change history:                                ed. 16/06/86
*/

#include "pumpdefs.c"
#include "pumpglob.c"
#include "monaux.c"
#include "errmes.c"
#include <pcs.h>

static direct PANPTR pumpfix;
static direct TIPTR timfix;
static direct TI ts;
int sigflag;

main () {
    char *inbu;
    int intrupt(),pa;
    unsigned ptr;
    double atof();

    timfix = &ts;
    intercept(intrupt);
    system ("tmode nopause");
    clrscrho();
    backinvers();
    system (MASK1);
    inversfor();
    if ((pa=open(PUMPAD,1)) == -1) {
        clmess();
        write (0,mess1,sizeof(mess1));
        alarm();
        exit (0);
    }
    read (pa,&ptr,sizeof(unsigned));
    pumpfix = ptr;
    close (pa);
    sigflag = 0;
    for (;;) {
        curback();
        readln (0,inbu,2);
        if ((*inbu != 'c') && (*inbu != 'd')) {
            clmess ();
            write(0,mess2,sizeof(mess2));
            sleep(4);
            clmess ();
            curback ();
        }
        if (*inbu == 'd') display (pumpfix,timfix);
        if (*inbu == 'c') change (pumpfix,timfix);
    }
}
```

```
/*----- display -----*/
display (a,t)
  register PANPTR a;
  register TIPTR t;
{
  int c,secs,minute,time;

  for (;;) {
    if (sigflag == 1) break;
    clfield (2,1,48); printf("%d\n",PUMPPROCS);
    clfield (2,3,13); printf("%d\n",getprstat (PUMPPROCS));
    clfield (4,3,32); printf("%d\n",a->record);
    time = a->nextsto;
    minute = abs(time/60);
    secs = time - (minute*60);
    clfield (6,3,60); printf("%02d:%02d\n",minute,secs);
    clfield (5,6,24); printf("%04d\n",a->meanint/60);
    time = a->nextmean;
    minute = abs(time/60);
    secs = time - (minute*60);
    clfield (6,7,65); printf("%02d:%02d\n",minute,secs);
    clfield (4,7,24); printf("%04d\n",a->horint/60);
    time = a->nexthormean;
    minute = abs(time/60);
    secs = time - (minute*60);
    clfield (6,7,65); printf("%02d:%02d\n",minute,secs);
    clfield (4,8,24); printf("%d\n",a->horlag/60);
    clfield (4,8,65);
    if (a->untilhoriba > 0) printf("%d\n",abs(a->untilhoriba/60));
    else printf("Int\n");
    clfield (4,9,24); printf("%d\n",a->changint/60);
    time = a->untilnext;
    minute = abs(time/60);
    secs = time - (minute*60);
    clfield (7,9,65);
    if (a->untilnext == -111) printf("SLEEP\n");
    else printf("%03d:%02d\n",minute,secs);
    clfield(3,11,67);
    if (a->transflag == ACTIF) printf("ON\n");
    else printf ("OFF\n");
    clfield(3,12,67);
    if (a->meanflag == ACTIF) printf("ON\n");
    else printf ("OFF\n");
    clfield(3,13,67);
    if (a->gasflag == ACTIF) printf("ON\n");
    else printf ("OFF\n");
    clfield(3,14,67);
    if (a->albinflag == ACTIF) printf("ON\n");
    else printf ("OFF\n");
    clfield(3,15,67);
    if (a->pumpflag == ACTIF) printf("ON\n");
    else printf ("OFF\n");
    clfield(3,16,67);
    if (a->plstatus == NOT ACTIF) printf("OFF\n");
    if (a->plstatus == ACTIF) printf("ON\n");
    distim (t);
    tsleep (150);
  }
  sigflag = 0;
}
```

```
/*----- change -----*/
change (a,t)
    register PANPTR a;
    register TIPTR t;
{
    char inbu[6];
    int c,secs,minute,time;

    for (;;) {
        if (sigflag == 1) break;
        clfield (2,1,48); printf("%d\n",PUMPPROCS);
        clfield (2,3,13); printf("%d\n",getprstat(PUMPPROCS));
        clfield (4,3,32); printf("%d\n",a->record);
        time = a->nextsto;
        minute = abs(time/60);
        secs = time - (minute*60);
        clfield (6,3,60); printf("%02d:%02d\n",minute,secs);
        clfield (5,6,24); printf("%04d\n",a->meanint/60);
        time = a->nextmean;
        minute = abs(time/60);
        secs = time - (minute*60);
        clfield (6,6,65); printf("%02d:%02d\n",minute,secs);
        clfield (4,7,24); printf("%04d\n",a->horint/60);
        time = a->nexthormean;
        minute = abs(time/60);
        secs = time - (minute*60);
        clfield (6,7,65); printf("%02d:%02d\n",minute,secs);
        clfield (4,8,24); printf("%d\n",a->horlag/60);
        clfield (4,8,65);
        if (a->untilhoriba >0) printf("%d\n",abs(a->untilhoriba/60));
        else printf("Int\n");
        clfield (4,9,24); printf("%d\n",a->changint/60);
        time = a->untilnext;
        minute = abs(time/60);
        secs = time - (minute*60);
        clfield (7,9,65);
        if (a->untilnext == -111) printf("SLEEP\n");
        else printf("%03d:%02d\n",minute,secs);
        clfield(3,11,67);
        if (a->transflag == ACTIF) printf("ON\n");
        else printf ("OFF\n");
        clfield(3,12,67);
        if (a->meanflag == ACTIF) printf("ON\n");
        else printf ("OFF\n");
        clfield(3,13,67);
        if (a->gasflag == ACTIF) printf("ON\n");
        else printf ("OFF\n");
        clfield(3,14,67);
        if (a->albinflag == ACTIF) printf("ON\n");
        else printf ("OFF\n");
        clfield(3,15,67);
        if (a->pumpflag == ACTIF) printf("ON\n");
        else printf ("OFF\n");
        clfield(3,16,67);
        if (a->plstatus == NOT ACTIF) printf("OFF\n");
        if (a->plstatus == ACTIF) printf("ON\n");
        distim (t);
        clfield (2,18,42);
        readln (0,inbu,2);
    }
}
```

```
switch (*inbu) {
case 'a': clfield(6,6,24);
        readln (0,inbu,4);
        a->meanint = atoi(inbu)*60;
        clfield (2,18,42);
        break;
case 'b': clfield(6,7,24);
        readln (0,inbu,4);
        a->horint = atoi(inbu)*60;
        clfield (2,18,42);
        break;
case 'c': clfield(6,8,24);
        readln (0,inbu,4);
        a->horlag = atoi(inbu);
        clfield (2,18,42);
        break;
case 'd': clfield(6,9,24);
        readln (0,inbu,4);
        a->changint = atoi(inbu);
        clfield (2,18,42);
        break;
case 'e': clfield(3,11,67);
        if(a->transflag == ACTIF) {
            a->transflag = NOT_ACTIF;
            printf("OFF\n");
        }
        else {
            system(TRANS);
            printf("ON\n");
        }
        clfield (2,18,42);
        break;
case 'f': clfield(3,12,67);
        if(a->meanflag == ACTIF) {
            a->meanflag = NOT_ACTIF;
            printf("OFF\n");
        }
        else {
            system(MEAN);
            printf("ON\n");
        }
        clfield (2,18,42);
        break;
case 'g': clfield(3,13,67);
        if(a->gasflag == ACTIF) {
            a->gasflag = NOT_ACTIF;
            printf("OFF\n");
        }
        else {
            system(GAS);
            printf("ON\n");
        }
        clfield (2,18,42);
        break;
case 'h': clfield(3,14,67);
        if(a->albinflag == ACTIF) {
            a->albinflag = NOT_ACTIF;
            printf("OFF\n");
        }
}
```

```

        else {
            system(GASALBIN);
            printf("ON\n");
        }
        clfield (2,18,42);
        break;
    case 'i': clfield(3,15,67);
        if(a->pumpflag == ACTIF) {
            a->pumpflag = NOT_ACTIF;
            printf("OFF\n");
        }
        else {
            system(PUMP);
            printf("ON\n");
        }
        clfield (2,18,42);
        break;
    case 'k': clfield(3,16,67);
        if (a->plstatus == NOT_ACTIF) {
            if ((a->pidplant = os9fork(PLANT,0,0,1,1,0,256)) == -1) {
                clmess();
                write (0,mess5,sizeof(mess3));
                sleep (3);
                clmess();
                clfield (2,18,42);
                break;
            }
            a->plstatus = ACTIF;
            printf("ON\n");
        }
        if (a->plstatus == ACTIF) {
            kill (a->pidplant,2);
            sleep (5);
            a->plstatus = NOT_ACTIF;
            printf("OFF\n");
        }
        clfield (2,18,42);
        break;
    } /* switch */
    inbu[0]='0';
} /* for(;;) */
sigflag = 0;
}

/*----- distim -----*/
distim (t)
    register TIPTR t;
{
    getdate (getsecs(),t);
    clfield (17,1,57);
    printf ("%2d/%02d/%02d ",t->year,t->month,t->day);
    printf (" %02d%02d:%02d\n",t->hour,t->min,t->sec);
}
/*----- intrupt -----*/
intrupt (sig) {
    if (sig == 2) {
        clscrho();
        backinvers();
        exit(0);
    }
    if (sig == 3) sigflag = 1;
}

```

```
/*  pumplnt.c
    PROZESSSTATUS auf FARBMONITOR
    purpose: display of pump control system on color screen
    programmer:      ed.      05. 05. 86.
    change history:

*/
#include "pumpdefs.c"
#include "pumpglob.c"
#include "errmes.c"
#include "monaux.c"
#include <graz.h>
static direct PANPTR pumpfix;

main () {
    char buffer[20],
          edrefres =0,
          adrefres =0;
    int pn;
    register PANPTR p;
    unsigned ptr;

    if ((pn=open(PUMPAD,1)) == -1) {
        clmess();
        write(0,mess1,sizeof(mess1));
        exit(0);
    }
    read (pn,&ptr,sizeof(unsigned));
    pumpfix = ptr;
    close (pn);
    grazini();
    allclr();
    title();
    dfilter(150,50);
    wastin(20,50);
    drawpump(114,44);
    drawph(195,175);
    drawgas(180,200);
    drawcomp();
/*  main loop */
    for (;;) {
        p = pumpfix;
        p->pumpval = 50.0;
        clrwindow (337,11,37,43);
        clrwindow (432,11,37,64);
        sprintf (buffer,"%4.3f\n",p->pHmean);
        drawstr (0,0,1,1,337,33,RED,buffer);
        sprintf (buffer,"%4.3f\n",p->hormean);
        drawstr (0,0,1,1,337,22,RED,buffer);
        sprintf (buffer,"%5.1f\n",p->tylmean);
        drawstr (0,0,1,1,432,33,RED,buffer);
        if (p->wpos == WPOS AD) sprintf (buffer,"ALBIN\n");
        else sprintf (buffer,"ETTORE\n");
        drawstr (0,0,1,1,432,22,RED,buffer);
        clrwindow (295,170,30,9);
        sprintf (buffer,"%2.3f\n",p->pHin);
        drawstr (0,0,1,1,295,170,RED,buffer);
        clrwindow (410,162,30,9);
        sprintf (buffer,"%5.1f\n",p->tylin);
        drawstr (0,0,1,1,410,162,RED,buffer);
        clrwindow (275,132,30,9);
```

```
    sprintf (buffer,"%3.1f%%\n",p->horin);
    drawstr (0,0,1,1,275,132,RED,buffer);
    clrwindow (128,21,23,9);
    clrwindow (85,12,50,9);
    if (p->pumpval == 0) {
        sprintf (buffer,"OFF\n");
        drawstr (0,0,1,1,128,21,RED,buffer);
    }
    else {
        sprintf (buffer,"ON\n");
        drawstr (0,0,1,1,128,21,RED,buffer);
        sprintf (buffer,"%4.1f ml/h\n",p->pumpval);
        drawstr (0,0,1,1,85,12,RED,buffer);
    }
    if (p->wpos == WPOS_AD)
        if (adrefres == 0) {
            gasalbin ();
            adrefres = 1;
            edrefres = 0;
        }
    if (p->wpos == WPOS_EDP)
        if (edrefres == 0) {
            gased ();
            edrefres = 1;
            adrefres = 0;
        }
        sleep (2);
    }
}
/*----- title -----*/
title () {
    char buffer[60];
    sprintf (buffer,"ANAEROBIC TREATMENT OF WASTEWATER");
    drawstr (0,0,2,2,50,235,WHITE,buffer);
    colatt (WHITE,0x01);
    rellin (0,0,233,512,0);
}
/*----- dfilter -----*/
dfilter(xpos,ypos)
int xpos,ypos;
{
    colatt (WHITE,0x01);
    rellin (3,xpos+2,ypos,0,150);
    rellin (3,xpos+60,ypos,0,150);
    rellin (2,xpos-4,ypos,68,0);
    rellin (2,xpos-4,ypos+150,68,0);
    rellin (0,xpos,ypos+130,-63,0);
    rellin (0,xpos-63,ypos+130,0,-22);
    rellin (0,xpos,ypos+132,-66,0);
    rellin (0,xpos-66,ypos+132,0,-24);
    colatt (GREEN,0x01);
    rellin (53,xpos+56,ypos+3,0,120);
    colatt (BLUE,0x01);
    rellin (53,xpos+56,ypos+121,0,11);
    rellin (0,xpos+3,ypos+131,-66,0);
    rellin (0,xpos-64,ypos+108,0,23);
}
```

```
/*----- wastin -----*/
wastin(xpos,ypos)
int xpos,ypos;
{
    colatt (WHITE,0x01);
    rellin (2,xpos,ypos,60,0);
    rellin (2,xpos,ypos,0,20);
    rellin (2,xpos+60,ypos,0,20);
    rellin (0,xpos+29,ypos,0,-7);
    rellin (0,xpos+32,ypos,0,-5);
    rellin (0,xpos+32,ypos-5,124,0);
    rellin (0,xpos+156,ypos-5,0,5);
    rellin (0,xpos+29,ypos-7,130,0);
    rellin (0,xpos+159,ypos-7,0,7);
    colatt (RED,0x01);
    rellin (56,xpos+57,ypos+2,0,17);
    rellin (0,xpos+30,ypos-6,128,0);
    rellin (1,xpos+31,ypos-6,0,7);
    rellin (1,xpos+158,ypos-6,0,8);
}
/*----- drawph -----*/
drawph(xpos,ypos)
int xpos,ypos;
{
    int x,y;
    char buffer[40];
    colatt (WHITE,0x01);
    rellin (4,xpos,ypos,0,40);
    rellin (0,xpos-2,ypos+40,0,5);
    rellin (0,xpos-2,ypos+45,105,0);
    rellin (0,xpos+103,ypos+45,0,-40); /* draw pH electrode and line */
    drawbox (258,165,80,15,1);
    sprintf (buffer,"pH:\x0");
    x = xpos+75;
    y = ypos-5;
    drawstr (0,0,1,1,x,y,WHITE,buffer);
}
/*----- drawgas -----*/
drawgas (xpos,ypos)
int xpos,ypos;
{
    char buffer[40];
    colatt (YELLOW,0x01);
    rellin (0,xpos,ypos,0,25);
    rellin (0,xpos,ypos+25,250,0);
    rellin (0,xpos+250,ypos+25,0,-45); /* Gasline */
    colatt (WHITE,0x01);
    drawbox (390,160,80,20,1);
    sprintf (buffer,"GASFLOW:\x0");
    drawstr (0,0,1,1,400,171,WHITE,buffer);
    drawbox (258,127,80,33,1);
    sprintf (buffer,"METHANE:\x0");
    drawstr (0,0,1,1,269,147,WHITE,buffer);
    drawbox (390,107,80,20,1);
    sprintf (buffer,"FERMENTER\x0");
    drawstr (0,0,1,1,399,113,WHITE,buffer);
}
```

```
/*----- drawcomp -----*/
drawcomp() {
    char buffer[40];
    colatt (WHITE,0x01);
    drawbox (250,9,228,81,1);
    rellin (0,250,76,228,0);
    sprintf (buffer,"AF pump-controlling program\x0");
    drawstr (0,0,1,1,275,79,RED,buffer);
    sprintf (buffer,"Next change pumpperformance :\x0");
    drawstr (0,0,1,1,257,66,WHITE,buffer);
    sprintf (buffer,"Next change gassource :\x0");
    drawstr (0,0,1,1,257,55,WHITE,buffer);
    sprintf (buffer,"Next storage: Records:\x0");
    drawstr (0,0,1,1,257,44,WHITE,buffer);
    sprintf (buffer," pH_m: flow_m:\x0");
    drawstr (0,0,1,1,257,33,WHITE,buffer);
    sprintf (buffer," %%CH4 m: source:\x0");
    drawstr (0,0,1,1,257,22,WHITE,buffer);
    sprintf (buffer," COD in: COD out:\x0");
    drawstr (0,0,1,1,257,11,WHITE,buffer);
/* lines to coputer-box */
    rellin (0,249,143,10,0);
    rellin (0,249,100,0,43);
    rellin (0,249,100,115,0);
    rellin (0,364,90,0,10); /* until here Horiba-line */
    rellin (0,239,173,20,0);
    rellin (0,239,95,0,78);
    rellin (0,239,95,121,0);
    rellin (0,360,90,0,5); /* until here pH-line */
    rellin (0,470,170,10,0);
    rellin (0,480,95,0,75);
    rellin (0,368,95,112,0);
    rellin (0,368,90,0,5); /* until here Tylan-line */
    rellin (0,154,20,91,0);
    rellin (0,245,20,0,63);
    rellin (0,245,83,5,0); /* until here pump-line */
}
/*----- drawpump -----*/
drawpump(xpos,ypos)
int xpos,ypos;
{
    int x,y;
    char buffer[40];
    x = 6;
    y = 0;
    colatt (WHITE,0x01);
    piecyc (220,107,xpos,ypos,x,y,&x,&y);
    colatt (BLACK,0x01);
    filtrl (xpos-4,ypos,6,-3,6,3);
    colatt (WHITE,0x01);
    drawbox (74,9,80,22,1);
    sprintf (buffer,"PUMP:\x0");
    drawstr (0,0,1,1,85,21,WHITE,buffer);
    rellin (0,114,31,0,8);
}
```

```
/*----- gased -----*/
gased () {
    colatt (BLACK,0x01);
    rellin (0,417,147,40,0);
    rellin (0,339,137,104,0); /* deletes old lines */
    colatt (YELLOW,0x01);
    rellin (0,416,159,0,-8);
    rellin (0,339,147,104,0);
    rellin (0,417,129,0,7);
    rellin (0,417,137,40,0); /* draws new gas-lines */
    colatt (WHITE,0x01);
    drawvent(414,145);
    drawvent(442,135);
    colatt (RED,0x01);
    drawvent(414,135);
    drawvent(442,145);
}
/*----- gasalbin -----*/
gasalbin () {
    colatt (BLACK,0x01);
    rellin (0,339,147,104,0);
    rellin (0,417,137,40,0); /* deletes old lines */
    colatt (YELLOW,0x01);
    rellin (0,416,159,0,-8);
    rellin (0,417,147,40,0);
    rellin (0,417,129,0,7);
    rellin (0,339,137,104,0); /* draws new gas-lines */
    colatt (RED,0x01);
    drawvent(414,145);
    drawvent(442,135);
    colatt (WHITE,0x01);
    drawvent(414,135);
    drawvent(442,145);
}
/*----- drawvent -----*/
drawvent (xpos,ypos)
    int xpos,ypos;
{
    rellin (0,xpos,ypos,5,0);
    rellin (0,xpos,ypos,5,5);
    rellin (0,xpos+5,ypos,-5,5);
    rellin (0,xpos,ypos+5,5,0);
}
/*----- drawstr -----*/
drawstr (direc,incl,xscal,yscal,curx,cury,color,text)
    int direc,incl,xscal,yscal;
    register int curx,cury;
    long color;
    register char *text;
{
    colatt (color,0x01);
    chaatt (direc,incl,xscal,yscal);
    setcur (curx,cury);
    while (*text != 0) outcha (*text++,0,0,0,0);
}
```

```
/*----- drawbox -----*/
drawbox (x,y,dx,dy,width)
  int x,y,dx,dy,width;
{
  rellin (width,x,y,dx,0);
  rellin (width,x,y+dy,dx,0);
  rellin (width,x+width,y,0,dy);
  rellin (width,x+dx,y,0,dy);
}
/*----- clrwindow -----*/
clrwindow (xstart,ystart,length,highth)
  int xstart,ystart,length,highth;
{
  colatt (BLACK,0x01);
  rellin (length,xstart+length,ystart,0,highth);
}
/*-----*/
```

7. Literaturverzeichnis

- [1] Isaac, P.C.G. :
Waste treatment.
Metcalf & Eddy (eds.), Waste water engineering. Mc Graw Hill, New York (1960).
- [2] Nyns, E.J. :
Digesters. A worldwide review.
1st Int'l Symp. Anaerobic Digestion, Cardiff, Wales (1979).
- [3] Parikh, J.K., Parikh, K.S. :
Potential of biogas plants and how to realize it.
Schlegel & Barnea (eds.), Microbial energy conversion, Pergamon Press (1976).
- [4] Zinder, S.H. :
Microbiology of anaerobic conversion of organic wastes to methane: recent developments.
Amer. Society for Microbiol. News **50**, 294-298 (1984).
- [5] Van den Berg, L. :
Developments in methanogenesis from industrial waste water.
Can. J. Microbiol. **30**, 975-990 (1984).
- [6] Schröpfer, G.J., Fuller, W.J., Johnson, A.S., Ziemke, N.R., Anderson, J.J. :
The anaerobic contact process as applied to packinghouse wastes.
Sewage Industrial Wastes **27**, 460-486 (1955).
- [7] Reinhold, D., Noack, W. :
Biogas in der Landwirtschaft.
Elsner Verlag, Darmstadt (1955).
- [8] Young, J.C., McCarty, P.L. :
The anaerobic filter for waste treatment.
Proceedings of the 22nd Purdue Industrial Waste Conference, Ann Arbor Science Publishers Inc., Ann Arbor MI, 559-574 (1967).

- [9] Lettinga, G., Van Welson, S.W., Hobma, W., De Zeeuw, W., Klapwyk, A. :
Use of the upflow sludge blanket (USB) reactor concept for biological waste water treatment especially for anaerobic treatment.
Biotechnology Bioengineering **22**, 699-734 (1980).
- [10] Jewell, W.J., Switzenbaum, M.S., Morris, J.W. :
Municipal waste water treatment with the anaerobic attached microbial film expanded bed process.
J. Water Pollution Control Fed. **53**, 482-490 (1981).
- [11] Van den Berg, L., Kennedy, K.J. :
Support materials for stationary fixed film reactors for high-rate methanogenic fermentations.
Biotechnology Letters **3**, 165-170 (1980).
- [12] Pypin, P., Verstraete, W. :
Lactate and ethanol as intermediates in two-phase anaerobic digestion.
Biotechnology and Bioengineering **23**, 1145-1148 (1981).
- [13] Ulrici, W.O. :
Zweistufige Schlammstabilisierung mit Schlammwäsche.
5. Symp. Technische Mikrobiologie, Berlin, 131-143 (1982).
- [14] *Standard methods for the examination of water & wastewater.*
12th Edition, Amer. Pub. Health Assn., New York, N.Y.(1965).
- [15] Brandl, H., Bachofen, R. :
Mikrobielle Prozesse unter oxidationsmittel-limitierten Bedingungen an der Sediment-Wasser-Übergangszone in Seen.
Dissertation Universität Zürich (1986) in Druck.
- [16] Dubach, A. :
Charakterisierung einer Reinkultur methanogener Bakterien.
Diplomarbeit, Universität Zürich (1981).
- [17] Hungate, R.E. :
A roll tube method for cultivation of strict anaerobes.
Methods in microbiology, Vol **3B**, Norris, J.R., Ribbons, D.W. (eds.), Academic Press, New York.

- [18] Schneider, K., Frischknecht, K. :
Determination of the O_2 and CO_2 $K_L a$ values in fermenters with the dynamic method measuring the step responses in the gas phase.
J. Appl. Chem. Biotechnology **27**, 631-642 (1977).
- [19] Muster, P., Binder, A., Schneider, K., Bachofen, R. :
Influence of temperature and pH on the growth of the thermophillic cyanobacterium Mastigocladus laminosus in continuous culture.
Plant & Cell Physiol. **24**(2), 273-280 (1983).
- [20] Gosh, S., Conrad, J.R., Klass, D.:
Anaerobic acidogenesis of wastewater sludge.
J. Water Pollution Control Fed. **47**, 30-45 (1975).
- [21] Tesch, W.A. :
Methanerzeugung und Abbau organischer Stoffe in einem Anaeroben Festbettreaktor (Anaerobfilter).
Diplomarbeit, Institut für Pflanzenbiologie, Universität Zürich (1981).
- [22] Tesch, W.A., Schneider, K., Bachofen, R.:
Performance of an anaerobic filter for purification and methane production from waste water of a sugar refinery.
Process Biochemistry **January/February**, 34-37 (1983).
- [23] Ramalho, R.S.:
Introduction to wastewater treatment processes.
Academic Press (1977).
- [24] Zoetemeyer, R.J., van den Heuvel, van Andel, J.G., Boelhouwer, C. :
Two-phase anaerobic digestion: development and application.
1st Int'l Symp. Anaerobic Digestion, Cardiff, Wales (1979).
- [25] Kernighan, B.W., Ritchie, D.M. :
Programmieren in C.
PC professionell, Carl Hanser Verlag Wien (1983).
- [26] de Vletter, R., van Gils, W. :
COD balances as a tool in the wastewater managment of sugar factories.
Zucker **29**, 53-57 (1976).

- [27] Bhatia, D., Vieth, W.R., Venkatasubramanian, K. :
Steady-state and transient behavior in microbial methanification: I. experimental results.
Biotechnology and Bioengineering, Vol XXVII, 1192-1198 (1985).
- [28] Cohen, A., Zotemeyer, R.J., van Deursen, A., van Andel, J.G. :
Anaerobic digestion of glucose with separate acid production and methane formation.
Water Research **13**, 571-580 (1979).
- [29] Suwarnat, K., Weyrauch, W. :
Anaerobe Behandlung kohlenhydratreicher Abwasser.
Wasser, Luft und Betrieb **22** (11), 572-575 (1978).
- [30] Asinari, C.M., Binot, R., Fripiat, J.L., Naveau, H.P., Nyns, E.J. :
Yields and rates of methane production by pilot-scale bioconversion of a typical waste: panch manure and a typical energy crop: fresh-water macroalgae.
4. Symposium Technische Mikrobiologie, Berlin, 201-212 (1979).
- [31] Bhatia, D., Vieth, W.R., Venkatasubramanian, K. :
Steady-state and transient behavior in microbial methanification: II. mathematical modeling and verification.
Biotechnology and Bioengineering, Vol XXVII, 1199-1207 (1985).
- [32] Genung, R.K., Million, D.L., Hancher, C.W., Pitt Jr., W.W. :
Pilot plant demonstration of an anaerobic fixed-film bioreactor for wastewater treatment.
Biotechnology and Bioengineering Symp. **8**, 329-344 (1978).
- [33] Aivasidis, A. :
Scale-up of a new fixed bed loop reactor for anaerobic digestion.
3rd European Congress on Biotechnology, Vol **3**, 51-60 (1984).
- [34] McCarty, P.L. :
Anaerobic process.
Presented at the Birmingham short course on design aspect of biological treatment, Int'l Assn. of Water Pollution Research, Birmingham, England (1974).

- [35] Chen, Y.R., Hashimoto, A.G. :
Kinetics of methane fermentation.
Biotechnology and Bioengineering Symp. **8**, 269-282 (1978).
- [36] Salkinoja-Salonen, M.S., Sutton, P.M., Van den Berg, L., Wheatley, A.D. :
Starting up of an anerobic fixed film reactor.
Proceedings of the Int'l Assn. on Water Pollution Research Seminar on
Anaerobic Treatment of Waste Water in Fixed Film Reactors, Copenhagen
(1982).
- [37] Gosh, S., Klass, D.L. :
Two-phase anaerobic digestion.
Process Biochemistry **April**, 15-24 (1978).
- [38] Messing, R.A., Stineman, T.L. :
Rapid production of methane with immobilized microbes.
Biochemical Engineering III. Annals of the New York Academy of Sciences
413, 501-513 (1983).
- [39] Huysman, P., Van Meenen, P., Van Assche, P., Verstraete, W. :
*Factors affecting the colonization of non porous and porous packing materials
in model upflow methane reactors.*
Biotechnology Letters **5**, 643-648 (1983).
- [40] Fynn, G.H., Whitmore, T.N. :
*Retention of methanogens in colonised reticulated polyurethane foam biomass
support particle.*
Biotechnology Letters **6**, 81-86 (1984).
- [41] Akermark, B. :
*The system for effluent treatment and measures taken to reduce the water
requirement in beet sugar factories in Sweden.*
Socker Handligar **27**, 1-22 (1975).
- [42] Ahlgrimm, H.J., Orth, H.W. :
Energiefluss in Biogasanlagen.
5. Symposium Technische Mikrobiologie, Berlin, 69-75 (1982).

- [43] Follmann, H., Märkl, H. :
pH-statisches Verfahren zur kontinuierlichen anaeroben Vergärung von hochbelasteten Abwässern zu Methan. 1. Entwicklung des Konzeptes.
Milchwissenschaft **34** (6), 347-350 (1979).
- [44] Märkl, H., Witty, W. :
Reaktionstechnische Aspekte der Methangärung am Beispiel der Vergärung von Penicillinmycel.
Vortrag anlässlich der Arbeitssitzung des Fachausschusses "Bioverfahrenstechnik" der VDI-Gesellschaft Verfahrenstechnik und Chemieingenieurwesen (GVC) am 29./30. Mai 1984 in Lindau.
- [45] Podrutzny, M.F., van den Berg, L :
Development of a computer control system for anaerobic methane-producing reactors.
Biotechnology and Bioengineering, Vol.XXVI, 392-393 (1984).
- [46] Capri, M.G., Marais, G.v.R. :
pH adjustment in anaerobic digestion.
Water Research **9**, 307-313 (1975).
- [47] McCarty, P.L. :
Energetics and kinetics of anaerobic treatment.
Advanced Chememical Ser. **105**, 91-107 (1971).
- [48] Zehnder, A.J.B. :
Ecology of methane formation.
In: R.Mitchell (eds., John Wiley & Sons, New York), Water pollution microbiology Vol. **III**, 349-376 (1978).

8. Verdankungen

Diese Arbeit entstand am Institut für Pflanzenbiologie der Universität Zürich, Abteilung Mikrobiologie.

Herrn Prof. Dr. R. Bachofen danke ich herzlich für sein Interesse, für sein grosszügiges Entgegenkommen in der Beschaffung und Bereitstellung der Labor - Infrastruktur, sowie für die angenehme und persönliche Freiheit, die ich im Laufe der Tätigkeiten in seiner Abteilung erfahren durfte.

Herrn Dr. K. Schneider als meinem fachlichen Betreuer bin ich für die unentbehrliche technische Beratung und Mithilfe dankbar. Durch zahlreiche Diskussionen verhalf er mir zu einem Verständnis der verfahrenstechnischen Grundlagen. Ferner danke ich ihm für die Einführung in die Programmiersprachen PASCAL und C.

Der Leitung der Zuckerfabrik Frauenfeld danke ich für die Möglichkeit die mir gegeben war, dort meine Versuche durchzuführen und für die grosse Unterstützung die ich während meiner Arbeit erhalten habe. Der Dank gilt auch allen Angestellten, welche immer hilfsbereit zur Stelle waren falls dies nötig war.

Der schweizerische Nationalfonds unterstützte diese Untersuchungen im Rahmen des Projektes Nr. 3.520-0.83.

Nicht zuletzt gehört mein Dank all meinen Kommilitonen, insbesondere A. Dubach in der Abteilung Mikrobiologie.

CURRICULUM VITAE

D a P r a, Ettore, Bürger von Lozzo di Cadore, Italien, geboren am 5. Februar 1956 in Lozzo di Cadore.

- 1971 - 1976 *Kantonsschule in Schaffhausen*
- 1976 *Matura Typus C in Schaffhausen*
- 1976 - 1986 *Studium der Biologie an der Universität Zürich*
- 1980 - 1981 *Diplomarbeit am Institut für Pflanzenbiologie unter der Leitung von Prof. Dr. R. Bachofen im Hauptfach Mikrobiologie*
- 1981 *Diplom in Mikrobiologie*
- 1982 - 1986 *Dissertation am Institut für Pflanzenbiologie unter der Leitung von Prof. Dr. R. Bachofen und Dr. K. Schneider*
- 1982 - 1986 *Assistent am Institut für Pflanzenbiologie der Universität Zürich*

Vorlesungen und Kurse wurden besucht bei folgenden Dozenten und Assistenten:

W. Arndt, G. Bächli, R. Bachofen, E. Batschelet, J. Biegert, M. A. Billeter, M. Birnstiel, H. R. Bosshard, E. Brun, H. Burla, P. S. Chen, Ph. Christen, C. D. K. Cook, A. Dübendorfer, A. S. Dreiding, P. Endress, C. H. Eugster, H. Fischer, K. Hanselmann, M. Hesse, H. R. Hohl, R. Humbel, J. Kägi, F. Knüsel, H. Kummer, W. Kündig, P. F. Meier, H. R. Oswald, R. Parish, R. Portmann, D. Rast, W. Reichart, H. Rieber, H. P. Ruffner, W. Schaffner, K. Schneider, U. Straumann, G. Wagnière, H. Wanner, R. Wehner, Ch. Weissmann, K. Wilson, A. Zehnder, V. Ziswiler.

